Dan Gookin develops a 6th sense for a utility program with real options...pg 15

DOS RESOURCE GUIDE

The PC Productivity Magazine

SEPTEMBER 1994, #17
$4.95
(CANADA $5.95)

10 DOS TIPS You Can Use Today!

Our Special Report

DOS AND HARDWARE

- ★ Use DRIVPARM to add on a floppy
- ★ Tracking down an Interrupt conflict
- ★ Adding memory to a computer
- ★ How DOS works through the BIOS

FREE QBASIC PROGRAM...pg 56

One False Step And You Could Blow It!

Batch File Menu With CHOICE – It's Easy...pg 39

NEW Column: Windows Seat

Run DOS Programs Under Windows With an Alternate DOS PIF...pg 74

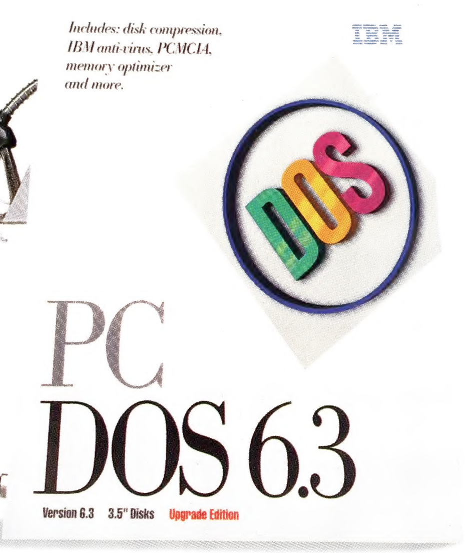
Add-in a QBasic Library of Sorting Subroutines...pg 62

AN IDG COMMUNICATIONS PUBLICATION

DEBUG tango: Rob Hummel makes your video dance with scripts that bounce...pg 30

High octane for your PC.

Want to pump up the performance of your PC? Fill 'er up with PC DOS 6.3. IBM's award-winning PC DOS is compatible with previous DOS versions, including MS-DOS.® But PC DOS 6.3 delivers more function, with an unmatched range of built-in utilities: IBM AntiVirus; integrated compression technology that's safe, fast and DoubleSpace® compatible; Central Point®'s full-screen backup including tape support; memory optimizer; full screen editor; CD-ROM support and more. And to top it all off, PC DOS will really get your Windows™ whirring.


Includes: disk compression, IBM anti-virus, PCMCIA, memory optimizer and more.
IBM
DOS
PC DOS 6.3
Version 6.3
3.5" Disks
Upgrade Edition

WHAT'S UNDER THE HOOD?	PC DOS 6.3	MS-DOS 6.21
Integrated compression.	YES	NO
Dynamic memory optimizer.	YES	NO
Number of viruses that can be detected.	2000	800
Full-featured backup support.	YES	NO
Program scheduler.	YES	NO
PCMCIA and Pen support features for mobile users.	YES	NO

With PC DOS 6.3 under the hood, you can shift your hardware *and* software into high gear.

To order or to find out more, call 1 800 342-6672.

In Canada, call **1 800 465-7999.** PC DOS 6.3 is also available at local software dealers.

IBM®

IBM is a registered trademark of International Business Machines Corporation. MS-DOS is a registered trademark and Windows is a trademark of Microsoft Corporation. All other products are trademarks or registered trademarks of their respective companies. © 1994 IBM Corp.

DOS RESOURCE GUIDE

EDITOR
Michael J. Comendul
SENIOR EDITOR
David Essex
ASSOCIATE EDITOR
Hilory Wagner
COPY/PRODUCTION EDITOR
Eileen Terrill
EDITOR AT LARGE
Eric Maloney
ARTICLE EDITORS
Patricia Cummings, Beth S. Jala,
Marilyn McMaster
TECHNICAL EDITORS
Hardin Brothers, John Wolfskill
CONTRIBUTING EDITORS
Dan Gookin, Robert Hummel,
Doug Lowe, Stan Miastkowski

ELECTRONIC PUBLISHING
Lindy P. Greenwood
DESIGN/PRODUCTION CONSULTANTS
Marilyn Fletcher, Dianne Ritson

PUBLISHER
Stephen Robbins
SALES REPRESENTATIVE
Linda M. Guyette
(603) 924-0325, (800) 343-0728
SALES REPRESENTATIVE
Laurie Donohue
(603) 924-0329, (800) 343-0728
LIST SALES REPRESENTATIVE
Christine Cahill
(508) 620-7718, (800) 343-6474

Business Computer Publishing Inc.

PRESIDENT
Jeff DeTray
EDITORIAL DIRECTOR
Michael J. Comendul
CIRCULATION MANAGER
Cathy Carabello
CIRCULATION ANALYST
Kim Smith
ADMINISTRATIVE ASSISTANT
Mary Hartwell
NEWSSTAND CIRCULATION
Kemco Publisher Services (603) 924-0224
CUSTOMER SERVICE
(800) 349-7327, (815) 734-1235

DOS Resource Guide (ISSN: 1056-7364) is published six times a year by Business Computer Publishing Inc., an IDG company. Business offices at 80 Elm St., Peterborough, NH 03458, (603) 924-0200. Entire contents copyright 1994 by Business Computer Publishing Inc. No part of this publication may be printed or otherwise reproduced without written permission from the publisher.

U.S. subscription rate is $23.70, one year; $44.00, two years; Canada (including GST) and Mexico, $29.70; foreign surface, $41.70, one year; foreign airmail, $53.70, one year. Payable in U.S. funds drawn on a U.S. bank. Prepayment is required on all foreign subscriptions. Second-class postage is paid at Peterborough, NH, and additional offices. POSTMASTER: Send address changes to *DOS Resource Guide*, P.O. Box 669, Mount Morris, IL 61054.

IDG COMMUNICATIONS *DOS Resource Guide* is a publication of International Data Group, the world's leader in information services on information technology.

PRINTED IN THE U.S.A.

EDITORIAL

IBM Broadens The DOS World

by Michael J. Comendul

A couple of IBM executives who dropped by the office recently with a copy of PC DOS 6.3 had more fun showing off an even newer PC DOS—something the company will label PC DOS 7. It will be outfitted with an OS/2 display called the Workplace Shell, an advanced interface that runs your computer by accessing your data and programs from any number of folders. The beta version is packed with stylish icons, attractive colors, and toolbar controls. In fact, so interesting was the demonstration—and so interesting was IBM's confirmation of a deal with Stac Electronics to tuck the compression utility Stacker into this next PC DOS—that the 6.3 upgrade faded from the conversation.

Why should speculation about a future product overshadow a new PC DOS? Because it's important news for readers of this magazine. An announcement like this reaffirms IBM's intention to stay in the DOS marketplace, even if Microsoft abandons it.

Then again, I can find still other reasons to overlook PC DOS 6.3. Its built-in compression utility (formerly the SuperStor product) is the most substantive add-in, but, after all, it was promised late last year when IBM released 6.1 without compression. And what are we to make of the Stac agreement? It seems a tacit admission from IBM that Stacker is the better compression utility, so why shouldn't users wait for PC DOS 7? (IBM denies that the company made the deal with Stac to avoid a copyright lawsuit.)

To its credit, version 6.3 continues to hone its feature edge, at least when its utilities are contrasted with MS-DOS's. An updated SMARTDRV will cache CD-ROM drives, and RAMBoost will now sense when you're using multiple configuration files and will optimize your memory for each configuration. In addition, IBM was never given credit for helping owners of older 8086 and 80286 computers gain up to 60K of extra application memory when they use the upper-memory drivers built into PC DOS 6.1. Add to this IBM's support for tape drives and PCMCIA cards and you see the company laying the groundwork for a stronger, more flexible DOS.

But notable as PC-DOS 6.3 might be, some add-ons seem overblown. My crib sheet tells me to mention that commas now appear in the numbers when you're displaying DIR, CHKDSK, FORMAT, and MEM. (I'll bet you can't wait.) PC DOS 6.3 plays catch-up with Microsoft in other ways: letting you use the hard drive as a medium for DISKCOPY, for example, ending the tedium of swapping "source" and "target" disks in drive A. Also, a new start-up feature lets you look at each line of your CONFIG.SYS and AUTOEXEC.BAT files as they execute, helpful when you're checking for an incompatibility.

IBM and companies like it are determined not just to keep the DOS marketplace alive, but to enrich it as well. Many readers may think that Microsoft defines the DOS market, but that's not true. In fact, *you* are the marketplace. You and your needs are much larger than Microsoft, and certainly more important.

This magazine, too, wants to stay a part of this broader world of DOS, and for that reason we'll be changing our name with the next issue. Gone is *DOS Resource Guide*; it's *DOS World* now. ■

DOS RESOURCE GUIDE

Number 17, September 1994

DOS

21 Tame Your Fussy Floppy Drives **by Hardin Brothers**
Bridge the communication gap: If you find yourself arguing with DOS over those stubborn storage devices, the commands DRIVPARM and DRIVER.SYS will help you settle your differences.

26 Solving IRQsome Problems **by Ken Johnson**
Plug-in cards and peripherals sometimes don't work because they try to talk to your computer on the same line. MS-DOS's Diagnostics program can help you find those "interrupt request" conflicts and locate available addresses for all your devices.

30 Put Some Life into Your Batch Files **by Robert L. Hummel**
You can get there from here. These short-and-sweet Debug scripts will set your cursor free, letting you design snazzy menus, simple animation, and whimsical special effects.

BAT

34 Turn Up the Volume **by Hardin Brothers**
You don't need to know a lot of intricate programming tricks to write clever and powerful routines. QBasic offers simple tools that can beef up DOS's batch-file language and let you check system information on the fly.

39 Pros' Choice: A Wealth of Options **by Doug Lowe**
The Dark Ages of menu programming are over, thanks to DOS 6's CHOICE command. Now you can create complex nested-menu routines that run from a single, interactive batch file.

FYI

44 Curing Short-Term Memory Maladies **by John M. Goodman**
Boosting your PC's megabyte quotient is the quickest, easiest way to get the most from your machine. But before you pick up that screwdriver, here's some practical advice on selecting the right kind of RAM and the right amount. ***Plus:*** *"Navigating the DOS Memory Maze" by Robert Gryphon.*

51 Understanding Your PC's BIOS **by John Bryan**
The BIOS is the all-important mediator between hardware and software—and knowing how it works will unlock your computer's deepest secrets and help you troubleshoot system failures.

BAS

56 Mad Bomber **by Doug Lowe**
An evil scientist has hidden 15 explosives within a 10-by-10 grid. Your mission, should you choose to accept it, is to find them fast—without blowing yourself to bits. You'll need all the deductive reasoning you can muster, plus a dash of old-fashioned luck. It's classic arcade fun.

62 Sorting It All Out **by Antonio and Alfonso De Pasquale**
Organizing lists can be a massive job. Storing information in QBasic arrays is only half the battle—these two sort modules will rearrange your data and keep your files on track.

On the Cover

Pictured on the cover is a CD-ROM-equipped 486/DX2 66MHz mini-tower from Blackship Computer Systems. Blackship offers an extensive range of custom-configured multimedia/networking systems, components, and peripherals.

COVER PHOTO BY LARRY DUNN

DRG

COLUMNS

1 Editorial **by Michael J. Comendul**
IBM expands the world of DOS.

15 Gookin on DOS **by Dan Gookin**
Basement programmers keep coming up with totally cool DOS utilities.

18 Reader Forum **by Hardin Brothers**
Printing QBasic graphics, SWITCHing on system security, fixing broken batch files.

68 Start-Up Clinic **by Jack Nimersheim**
Many computer owners suffer from the same malady: power envy.

74 Windows Seat **by Eric Maloney**
Create a special icon to run DOS commands while conserving precious memory.

88 Shareware Exchange **by Hardin Brothers**
Creating attractive batch-file menus.

DEPARTMENTS

4 Letters to the Editor
9 Tips from Readers
12 Best of the Batch
66 Readers' Queue
70 Q&A **by John Wolfskill**

TIPS

DOS

24 Too Many Cooks Spoil the Cache
Save memory and improve SmartDrive performance by cutting the number of buffers.
by John Wolfskill

25 Systematic Trash Removal
This batch file monitors the ages of your backup documents and purges your hard disk of the oldest ones automatically.
by Ronald L. Vanover

35 Look Before You Defragment
Grab those files you want to save before optimizing your hard drive.
by Herbert M. Dye, Jr.

48 A Temporary Fix
Setting up a special directory for temporary files relieves some of your hard-disk clutter.
by Larry Peace

53 Cut It Out!
This neat trick lets you delete a single line from a text file—right at the DOS prompt.
by John Wolfskill

61 The Sky's the Limit
Creating subdirectories on a floppy disk expands the number of files it can hold.
by John Wolfskill

65 A Well-Kept Path
Let the backslash do the work in your PATH statements—eliminate C: and save room.
by Lane Olinghouse

WINDOWS

20 How Much RAM Does a RAM Hog Hog?
Now you can calculate just how much memory that fancy wallpaper is costing you.
by John Wolfskill

33 Automatic Document Duplication
The Save As option on Write's File menu gives you a Backup box that'll update your duplicate right along with the original.
by Lane Olinghouse

42 Group Mentality
To boot programs and documents fast, collect related files in "work groups."
by Lane Olinghouse

How to Contact The Editors

DOS Resource Guide *welcomes letters, complaints, and submissions from readers. The easiest way to reach the editors is the U.S. mail:* DOS Resource Guide, *80 Elm St., P.O. Box 802, Peterborough, NH 03458-0802. All letters to the editor and questions are understood to be submitted for publication unless otherwise indicated. You can reach our staff electronically over CompuServe at 75300,2361. Please include your complete address and a daytime phone number on your correspondence.*

Also, you can reach the editors through the DRG *bulletin board system (603-924-3181). To connect, set your modem and software to 8 data bits, no parity, 1 stop bit.* DRG*'s BBS lists all the QBasic and shareware programs mentioned in these pages. Shareware items listed on the* DRG *BBS are products protected by copyright law. You're welcome to try these programs. If you find them useful, we ask you to register and pay the applicable fees to the programs' respective owners.*

DRG *encourages the submission of DOS tips, QBasic programs, and batch-file articles from readers. Please submit all material on disk in ASCII format together with a double-spaced printed copy to our offices. The magazine cannot be responsible for the return of these manuscripts, unless an appropriate mailer and return postage are enclosed.*

DRG *makes every effort to verify its articles and programs, but the magazine assumes no responsibility for any damages because of errors or omissions in our articles. Corrections to listings or articles are printed as soon as possible.*

DOS RESOURCE GUIDE

Letters to the Editor

When Time Stands Still

The problem you mentioned in your May 1994 editorial ["The People We Hear From," *DRG*#15, page 5] was discussed in *PC World's* November 1993 issue. It's called the Midnight Bug. By the way, I'm a great fan of *DRG*. Please pass on my undying gratitude to Hardin Brothers, from whom I've learned more about my computer than from all other sources combined. Dan Gookin is also a treasure.

Brian Smithers
Orlando, Fla.

Our sister publication, PC World, *wasn't the only magazine to describe the date problem that arises when 286 systems are left on all night.* DRG *also discussed this bug in a recent "Q&A"* [DRG*#14, March 1994, "Hardware," page 76*]. *The* DRG *BBS (603-924-3181; 8-N-1) offers a neat shareware fix: RightTime 2.5 (RITM25.ZIP).* *—Eds.*

Compression on a Shoestring

JAMMIT.BAT ["A Poor Man's Data Compression," *DRG*#14, March 1994, page 39] works nicely, even with floppies. But you don't need the latest version of PKZIP. It works under version 1.1, which is the only version I own. Because I'm a "poor man," I couldn't see spending $75 for an updated version.

Dale Patten
Topeka, Kan.

A WIN-DOS Situation

I'm new to computing, and *DOS Resource Guide* has been a great help to me, both with DOS and Windows. I've debated whether to renew my subscription, as you seem to have dropped Windows completely. I like articles that cover how to get the most out of both. Why not publish *Windows Resource Guide*?

Andrew J. Magne
Jacksonville, Fla.

This issue marks the début of "Windows Seat," a new column of tips and techniques by Eric Maloney, DRG*'s editor at large.*

In addition, look for our brand-new special issue, Maximize: The Practical Guide to Windows, *on newsstands this fall. It's loaded with solutions, hints and tricks, hands-on projects, Q&A, news, and commentary that will make computing with Windows both fun* and *productive.* *—Eds.*

An Easier Way

In discussing COPYCMD's switches ["Copy Minus the Clobber," *DRG* #13, January 1994, page 18], columnist Dan Gookin says that whenever you use the COPY, XCOPY, or MOVE command within a batch file, DOS assumes the /Y switch and overwrites any file of the same name as the one you're copying or moving. As a solution, he proposes including the /–Y switch whenever you invoke these commands within a batch file.

This seems unnecessary. If you set the COPYCMD environment variable to /–Y, COPY, XCOPY, and MOVE commands executed from batch files don't clobber existing files. Instead, they prompt you for permission before overwriting the files. DOS's on-line documentation (type `HELP SET` at the DOS prompt) explains this clearly.

The results of my tests indicate that Microsoft, not Gookin, is right on this point.

Paul Castonguay
Swampscott, Mass.

You're correct, of course, but I'm not incorrect. As I stated, if you don't specify /–Y *when you use* COPY, XCOPY, *or* MOVE *in a batch file, DOS clobbers files. You can specify* /–Y *by adding this switch to your batch-file commands or by setting the variable* COPYCMD.

—Dan Gookin

Technology Marches On

I was impressed when I read Hardin Brothers' statement in *DRG*#14, the March 1994 issue ["The Batch-File Mouse That Roared," page 43]: "Technology has a way of pushing us all forward, whether we want to go its way or not."

From my earliest programming (on an old Timex Sinclair), I've viewed computers as "mind tools." But today's popular trend is to view them as appliances, and PC users who treat them as such are the losers.

Yes, I do Windows occasionally, but only when I must use applications such as a CD-ROM encyclopedia or a multimedia program. Technology has pushed me that far. And vendors of graphical software laugh all the way to the bank as the billions pour in.

Yet DOS remains, and thank goodness it does. It's like an old friend to whom I can return for answers.

Why? Because I can program it to do what I want it to do.

Wendell R. Henry
Kerrville, Texas

Count Me Out

I purchased your February 1994 issue [*Running DOS and Windows in Harmony*] hoping to find some tips and hints about peaceful coexistence between DOS and Windows. Much to my surprise, the first piece I read ["Windows Is No Clear Choice," page 4] was Windows bashing at its worst.

Like other devout DOS users, I was dragged into Windows kicking and screaming. Undaunted by my threats, "Windows worms" poured disk after disk of Windows data onto my hard disk.

After a few seconds of staring blankly at my "new" screen, I then decided to get some work done. After a couple of "General protection fault" errors, I decided to call it quits for the day.

I returned to work the next day with a new attitude: I would learn Windows and like it. I started simply, then went on to more complex things. I discovered OLE (object linking and embedding), cutting, pasting, and myriad other Windows functions.

Before I knew it, I was asking for more Windows applications. Windows opened up a whole new world of productivity, enjoyment, and entertainment for me.

Statements by Editor at Large Eric Maloney are pure baloney. I'll grant that Windows's File Manager can't handle some tasks—MD, CD, RD, and DEL—as easily as DOS can, but the learning curve for Windows and Windows applications isn't steep. He must have coordination problems if he can't control his mouse well enough to click on an option button or check box.

As a result of your recent bout of Windows bashing, I will never purchase another issue of *DRG*.

Gregory H. Ogelthorpe
Lewisville, Texas

Asleep at the Wheel?

As a frequent *DRG* reader, I was appalled to see an issue devoted solely to Windows, or "Windoze," as I prefer to call it [*Running DOS and Windows in Harmony*, February 1994].

"Windoze" does have certain advantages over a straight DOS environment, but other software can also provide those advantages. Quarterdeck Office Systems' DESQView lets you run several programs at once and transfer data among programs. Even Central Point Software's PC Tools provides multitasking and memory management at a fraction of the toll in hard-disk space that "Windoze" exacts.

Never before have users been faced with the pressing need to upgrade their computers so that they can accommodate a program that provides little more than an interface for operating the latest programs. I hope that *DRG* doesn't intend to make "Windoze" a major focus of future issues.

Ian Scott
London, Ontario
Canada

OMISSIONS AND CORRECTIONS

The Truth of the Matter

I read your February 1994 issue [*Running DOS and Windows in Harmony*] because I've decided to buy a PC after owning a Macintosh for eight years. One reason for the switch is that more entertainment software is available for the PC.

One of my top hardware considerations is hard-disk size, and two of the articles in this issue present conflicting facts on the amount of hard-disk space DOS and Windows consume. Ann Fischer Lent ["Investing in New Hardware," page 13] says that MS-DOS 6.x and Windows 3.2 take nearly 25MB. Major Windows applications need 12MB to 15MB, she says, so I should expect a basic setup consisting of a word processor, spreadsheet program, database manager, and utilities to take 50MB. If this is true, a 200MB hard disk is down to 125MB before anything else is installed.

But then in "Streamlining Directories" [page 65], Doug Lowe says that the latest incarnation of MS-DOS, version 6.2, uses about 7MB, while Windows 3.1 uses 10MB. I'm confused.

Glen C. Garrett
Montrose, Calif.

We apologize for supplying conflicting information. To set the record straight, we consulted Microsoft. According to a company representative, you need 4MB of hard-disk space to install a complete version of MS-DOS 6.2. For those with 386 or later systems, a full implementation of Windows 3.1 requires 10MB of disk space. By trimming unnecessary features judiciously, you may reduce that to just over 6MB. Of course, you may also trim DOS 6.2 by deleting utilities you don't need.
—Eds.

Different Strokes For Different Versions

The procedure for modifying QBASIC.EXE outlined in the DOS tip "Open Says Me" (*DRG*#15, May 1994, page 55) won't work in MS-DOS 6.2. If you have this DOS version and want to modify QBASIC.EXE so that the Edit program displays all files instead of just those with a TXT extension, follow this procedure instead:

1. Make a backup copy of QBASIC.EXE.
2. Using a text editor, such as The Norton Utilities' Disk Edit, find the ASCII character string *.TXT in the QBASIC.EXE file. This corresponds to the hexadecimal sequence 2A 2E 54 58 54. In the version of QBASIC.EXE dated 9/30/93, this sequence occurs at offset 187,504.
3. Change the above-mentioned hex sequence to 2A 2E 2A 00 00.
4. Save the changes, and exit the disk editor.

The same tip also contained information for changing the default BAS extension QBasic uses. With DOS 6.2, you can't use the above method to make this change. *—Eds.*

Bad Command or Filename

I've often used James L. Kelly's write-protection tip ["Tips from Readers," *DRG*#14, March 1994, page 9], although I insert a space between ATTRIB and +R. I also add a few extra lines that put me into the root directory and include a /S switch to write-protect all my EXE and COM files:

```
:
CD\
ATTRIB +R AUTOEXEC.BAT
ATTRIB +R CONFIG.SYS
ATTRIB +R *.EXE /S
ATTRIB +R *.COM /S
```

David A. Youck
Onarga, Ill.

You're right. DOS demands that a space separate ATTRIB and its attribute. Sorry about the typos.

—Eds.

PRINTing Error

Although I had no idea when I might need to use the information in Doug Lowe's tip "You Can Print It from There" [*DRG*#14, May 1994, page 43], I decided to see whether I could create a PRN file from a Windows application and then use DOS's PRINT command to print it.

I read and reread the tip, but, try as I might, I couldn't get my printer to respond correctly. When I looked up PRINT in DOS's on-line help system, I discovered that its /B switch didn't seem to refer to binary switches.

But then I found that COPY had a /B and that you can use the command to direct a file to a printer port. The command I got to work is as follows:

```
COPY /B filename.PRN LPT1
```

Did Lowe mean to use the COPY command instead, or can PRINT really send PRN files to a printer?

Donald J. Wilson
Canton, Mich.

I apologize for the confusion. I did, of course, mean to use COPY rather than PRINT to print the binary file.
—Doug Lowe

Shareware Update

BACKINFO, the shareware data-recovery package developed by Chief Data Recovery Ltd. of Israel and mentioned in "Desktop Publishing for Free" ("Shareware Exchange," *DRG*#14, March 1994, page 80), has been upgraded and now has a U.S. distributor.

If you try the product and like it enough to become a registered user, contact EliaShim Microcomputers, 4005 Wedgemere Drive, Tampa, FL 33610, 800-477-5177 or 813-744-5177. The company will ship you PC RESC-U, a more versatile version of BACKINFO.

Although the retail price for PC RESC-U is $39, EliaShim will honor the $25 fee mentioned in "Shareware Exchange." For further information, see the PC RESC-U entry in our recent "2nd Annual Guide to Utilities," *DRG*#16, July 1994, page 53. *—Eds.*

Trouble Ahead

I encountered a problem while trying to use a control character (^) when naming a DOSKEY macro, as suggested in "10 Reasons to Load DOSKEY" [*DRG*#13, January 1994, page 26]. When I'm working at the DOS prompt and press the Ctrl key plus a letter to produce a control sequence such as ^E (in the command DOSKEY ^E=DIR /OE /P, for example), I can type Ctrl-E later and press Enter to view a sorted directory one page at a time. If I try to save DOSKEY macros to a file and then edit them as suggested in the article, however, DOS doesn't save the control character correctly. As a result, macros with names such as ^E and ^A don't work when you invoke them from within a macro batch file.

This technique does work, however, if I fire up DOS's editor, Edit, and use Ctrl-P to insert a proper control character. Another alternative, one that works from the DOS prompt, is pressing Alt and a three-character number corresponding to the ASCII value of a letter. Pressing Alt while typing 001, for instance, produces ^A.

On another topic, it would be nice if you provided an E-mail address to which readers could send comments.

Stephen Decarie
Calgary, Alberta
Canada

Sorry we weren't more explicit on how to create the control character. We're passing along your letter for the benefit of other readers who might have had this problem.

Regarding your other suggestion, we recently began publishing our electronic addresses on the magazine's table of contents. Perhaps you missed them, because they're in fine print. You may reach the editors on CompuServe (75300,2361) or via the DRG *BBS (603-924-3181). The BBS operates 24 hours per day; its settings are eight data bits, no parity, one stop bit (8,N,1).* *—Eds.*

REVOLUTIONARY MULTIMEDIA SOFTWARE

IT'S FREE

IT'S AT&T

RIO • THE HEART OF THE MULTIMEDIA CREATION COLLECTION

1-800-892-8549

The graphics and illustration program known for its **POWERFUL** *high-end capabilities just got better. The new RIO achieves* **FAST** *screen redraw and addresses large color palettes without the need for high-priced video cards. Now we've made it* **EASY** *to express your creative whims and visions using intuitive tools and hundreds of PostScript fonts. When you've finished your masterpiece, you can output to many popular file formats. The revolutionary RIO is at the heart of a feature rich* **MULTIMEDIA** *collection that includes Panorama and RIO Animator.* **SEE FOR YOURSELF**-*Try the new RIO FREE for thirty days. If you are not totally convinced of its value, simply return it. Keep it and we'll bill your credit card or your company purchase order $99. This spectacular* **FREE OFFER** *is limited to the first 50,000 responses-so* **ACT NOW.**

AT&T

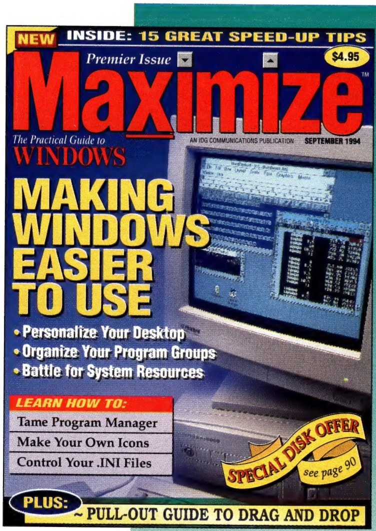
Circle 103 on Reader Service Card

Announcing:

A New Magazine Devoted to Windows Users

ON SALE SEPTEMBER 20th

YOU ASKED FOR IT: NOW, YOU'RE GOING TO GET IT!


NEW
INSIDE: 15 GREAT SPEED-UP TIPS
Premier Issue
$4.95
Maximize
The Practical Guide to WINDOWS
AN IDG COMMUNICATIONS PUBLICATION
SEPTEMBER 1994
MAKING WINDOWS EASIER TO USE
• Personalize Your Desktop
• Organize Your Program Groups
• Battle for System Resources
LEARN HOW TO:
Tame Program Manager
Make Your Own Icons
Control Your .INI Files
SPECIAL DISK OFFER
see page 90
PLUS:
~ PULL-OUT GUIDE TO DRAG AND DROP

WHY ARE THE EDITORS OF *DOS RESOURCE GUIDE* STARTING A WINDOWS MAGAZINE?

Because our phones are ringing off the hook!

Since *DOS Resource Guide* produced its first special issue about Windows, users have been calling us, asking for more information about how to set up and use Windows.

The result is **MAXIMIZE**, The Practical Guide to Windows.

In every issue, experts will answer your most frequently asked questions about this complex operating environment:

- How do I create a program group?
- Where do I find my icon files?
- My display is a mess: How can I get control of all of these programs, groups, and Windows?
- I installed a sound card; now, how do I make it work?
- With 8MB of memory, why do I still get "out of memory" messages loading three of my favorite programs?
- Why do I lose mouse control of my DOS programs under Windows?
- How can I adjust an INI file to help Windows find executable programs that don't fit my DOS PATH statement?

A lot of answers for only $5.95!*

Coming in September to a newsstand near you!

MAXIMIZE the Possibilities!... Call 1-800-349-7327

Reserve your copy now ($5.95* ppd.). Have your credit card ready. (Request Item #M94–)

*In Canada and Mexico, pay $6.95. Foreign Surface, $8.00. Foreign airmail, $13.00. Checks and money orders must be payable in U.S. funds drawn on a U.S. bank. American Express, VISA, MasterCard & Discover also accepted.

PDRG/001/AMX17

DOS RESOURCE GUIDE

Tips from Readers

Screen Blanking Made Simple

I've had my share of trouble with the screen savers I've tried.

For instance, because some of them don't monitor a mouse or a joystick, my screen sometimes used to go blank while I was playing a game. Even worse, pressing the appropriate key sequence sometimes didn't even bring my program back.

I solved the problem by writing a QBasic program, BLANK.BAS, and a small batch file, BLANK.BAT, that let me blank my screen on demand—usually when I want to leave the computer on, but expect to be away from my desk for an extended period. Here's BLANK.BAT:

```
@ECHO OFF
DIR |FIND /I "DIRECTORY OF" >
  C:\UTIL\BLANK.DAT
QBASIC /RUN C:\UTIL\BLANK.BAS
```

BLANK.BAS isn't much longer:

```
CLS
OPEN "C:\UTIL\BLANK.DAT"
  FOR INPUT AS #1
INPUT #1, OLDDIR$
CLOSE #1
OLDDIR$ = "CD " + MID$(OLDDIR$,
  14, LEN(OLDDIR$))
WAITFORKEY$ = INPUT$(1)
SHELL OLDDIR$
SYSTEM
```

I store these two short-but-handy files in my C:\UTIL directory, which is listed in my AUTOEXEC.BAT file's PATH statement.

When I want to blank the screen, I just type the word `BLANK` at the DOS prompt. BLANK.BAT writes the current directory information to a data file, BLANK.DAT, and runs BLANK.BAS, which clears the screen until I press a key.

BLANK.BAS uses the information in the data file to return me to the correct directory.

Jason Cannon
Elizabethtown, Ky.

If you want to accomplish the same end, but more efficiently, try this two-line version of BLANK.BAT:

```
@ECHO OFF
QBASIC /RUN C:\UTIL\BLANK.BAS
```

In addition, you may reduce BLANK.BAS to just three lines:

```
CLS
WAITFORKEY$ = INPUT$(1)
SYSTEM
```

—Eds.

An Ounce of Protection

If you "zip" and password-protect sensitive files, then delete the original files from your hard disk, the files aren't really protected. Anyone with even a passing knowledge of UNDELETE can bring the original data back to life.

If your nosy coworkers are savvy enough to understand UNDELETE, but unskilled in the ways of disk editors (such as The Norton Utilities' Disk Editor), however, you can give your files extra security without going to a lot of trouble.

After zipping a file but before deleting it, change its contents by storing a single word in the file. You may do this from a batch file or by typing an ECHO command at the DOS prompt. For instance, to protect the file MY-INCOM.TXT, type this command:

```
ECHO BYE>MY-INCOM.TXT
```

Thereafter, anyone who undeletes the file will see just a single word: *BYE.*

Robert J. de Vlam
Ouderkerk / Amstel
Netherlands

Period Piece

If you've been around DOS awhile, you probably know you can move from a subdirectory to its parent directory simply by typing `CD` followed by two periods. But the double periods are useful for many other situations, as well. You may use them with COPY or XCOPY to copy files to the directory one level above (type `COPY C:\WP\DATA.TXT ..`) or with DIR to view a directory one level above (`DIR ..`). If you experiment, you'll undoubtedly find other ways to make directory navigation shorter, simpler, and faster.

Mark J. Kingsbury
Battle Creek, Mich.

Ditto

Like many users, I store my data files one level below the directory containing the programs with which the files are associated.

To work from the data directory, while maintaining access to the program directory *and* keeping PATH short, I use a statement such as the following:

```
PATH .. \;C:\;C:\BAT;C:\DOS
```

I start each application by invoking a batch file that changes me to the correct disk and directory for the application's data files.

Because I include double periods (..) and a backslash (\) in PATH, the batch file can successfully invoke program files in the directory one level above the data-file directory.

Chris D. Wendt
Dickson,
Australian Capital Territory
Australia

Finders Keepers

The article "Golden Retrievers" [*DRG*#14, March 1994, page 48] prompts me to send this filename-finder macro, which I use in my start-up batch file:

```
DOSKEY FINDER=DIR C:\$1*.*/P/A/S/O
```

The sequence *.* lets me type just part of the filename for which I'm looking. For example, typing `FINDER AUTOE` helps me locate all versions of my AUTOEXEC file, and typing `.DRG` locates any files with the extension .DRG. The switch sequence shown above, /P/A/S/O, locates all files on my disk (even hidden ones) that meet my search criteria.

Michael Pryer
Toronto, Ontario
Canada

This method succeeds in finding files with three-character extensions because DOS truncates extensions if you supply too many characters. As a result, it interprets a sequence such as .DRG.* as .DRG. (When you substitute .DRG for $1, you produce the sequence .DRG*.*.)*

For the same reason, the macro lets you find files with extensions of two characters. (.DR becomes DR.*, which is interpreted as DR*.) You can't find a file by specifying a one-character extension, such as .D, however, because DOS would truncate it to .D..* —*Eds.*

Even Cleaner Comments

Kjell Martinsen offers a clever tip for using GOTO and labels to insert remarks into batch files and thus avoid slowing down processing ["Cleaner Comments," *DRG*#15, May 1994, page 47].

But you may also achieve this by using an undocumented feature of DOS. Instead of inserting REM, a space, and a comment, use a double colon (::), a space, and a comment, as follows:

```
:: Created on 12/10/93
```

Because DOS skips over lines beginning with a double colon, this technique lets you include remarks in your batch files, without slowing down program execution—*and* you won't have to keep track of multiple GOTO statements and labels.

Dan Shackelford
Fullerton, Calif.

Terminating TSRs

If, like me, you use a menu program, launching a memory resident program [a terminate-and-stay-resident program, or TSR] from the menu is simple. The tricky part is removing the TSR without exiting the menu or using a "mark and release" program to take a snapshot of your computer settings and to return the system later to that state.

I've come up with a solution that works with TSRs that use Alt-, Ctrl-, or Shift-key combinations to unload themselves. It involves editing the batch files the menu system uses to launch programs and TSRs.

For instance, suppose that selecting item 1 on your menu triggers a batch file that invokes a TSR for intercepting inbound fax calls and receiving faxes without your intervention. Make item 2 the option for removing the TSR cleanly:

```
@ECHO OFF
C:
CD \FAX
CALL FINI2.BAT
CD \
MENU.BAT
```

FINI2.BAT should contain instructions on which key combination to press to remove the fax TSR; it should end with a PAUSE command, as this example batch file does:

```
ECHO To terminate the fax,
  press Alt-H now
ECHO (hold down the Alt key
  and press H).
ECHO When the fax program
  ends, you will be
ECHO returned to the
  main menu.
PAUSE
```

Including PAUSE suspends processing until you press a key. If you press the right key sequence, which you should if the instructions are clear, PAUSE's request is satisfied, the fax program is terminated, and control returns to the batch file that set up the fax.

That batch file switches you to the root directory and then invokes the menu program, MENU.BAT.

I'm sure there are other neat workaround solutions to the problem, but this is the one I prefer.

Jerry Schlanser
Walton, Ky.

We're Looking for a Few
GOOD TIPS!

Have you ever discovered a new way of using a DOS command or written a clever batch file to perform a repetitive task? Sure you have! And *DOS Resource Guide* wants to publish your secrets. Tell us what your tip does and how readers might apply it.

All tips must be submitted on disk or electronically via

CompuServe 75300,2361
MCI Mail 668-4855
or the *DRG* BBS (603) 924-3181
Send your best DOS tips to:

DOS Resource Guide
80 Elm St., PO Box 802
Peterborough, NH 03458-0802

PowerBASIC: DOS Development
PC MAGAZINE EDITORS' CHOICE
• Instant TSRs
• Built-in Assembler
• Unsigned Integers, TYPES & UNIONS
• COM1-COM4 115200 BAUD, any IRQ
• Unlimited Dynamic String Space
• Array Sort/Scan/Insert/Delete
• BCD Variables
• Bit Operations
POPUP KEY CHR$ (1,41)
DO
POPUP SLEEP USING EMS
POPUP STUFF "^", 0, 0
LOOP
Create an instant TSR in just 5 lines of code.
PowerBASIC
We put the POWER in BASIC!
BASIC Compiler with Integrated Debugging Environment

"... DOS (BASIC) developers would do better to look beyond Microsoft.™"
PC Magazine, *September, 1993.*

PowerBASIC Awarded Editor's Choice

"These are tough criteria, and almost every product we looked at came up short. But not PowerBASIC, Version 3.0. Our Editor's Choice in the DOS category, it delivered in almost every regard. Its impressive feature set includes a wide selection of data types (including user-defined), support for unlimited string space, and the ability to create and manipulate large bit arrays. Its powerful editor and debugger make for fast development, and its in-line assembler is unique to the reviewed products. The fast compiler generates highly efficient executables for a wide variety of CPU/FPUs. The package also supports TSR Development. While price may not be a major issue for a professional development tool, the $149 price can't help but be attractive, and the product's documentation and support are exemplary." PC Magazine, September, 1993.

Move Up to . . .

PowerBASIC 3.0

Struggling with slow interpreters like QBasic or BASICA?™ Unhappy with bugs and outdated performance of other compilers? Then it's time to upgrade your code to PowerBASIC™ 3.0! Source code is 99% compatible with QuickBasic, so the move up to PowerBASIC couldn't be easier. Blazing speed ... precise calculation ... and a myriad of advanced features make PowerBASIC 3.0 the upgrade of choice.

Instant TSRs

Create Pop-Up TSRs with just five lines of code. A 600K TSR can "pop-down" to a memory image of just 4K, with automatic memory swapping to EMS or virtual disk. Plus, two-way communication between multiple TSRs is offered.

Built-in Assembler

Hand optimize critical sections of your code for absolute performance. Whether 1 line or 1000, mix BASIC and Assembler — the decision is yours.

Advanced Data Types

Unsigned integers (byte, word, double word). User-defined TYPES and UNIONS. Quad-word, 64-bit integers. Extended precision, 80-bit floats. Dynamic, fixed-length, and flex strings.

BCD Variables

Eliminate round-off errors, even those with currency variables. You aren't forced to use 4 decimal places. Rather, choose from 0 to 18 digits — always with automatic rounding to the precise level needed.

Serial Communications

COM1 through COM4. BAUD rates to 115200. Non-standard IRQ lines are all supported.

Advanced Features

Unlimited dynamic string space. Array Sort, Scan, Insert, Delete. Procedural floating point option for fast non-8087 performance. BIT Shift, Rotate, Test, Set, Reset, Toggle. Implied bit arrays up to 512K bits. Conditional compilation ($IF, $ELSE, $ENDIF). Text viewports. EXIT FAR allows multi-level sub/function exit (beats the LONGJMP in C, any day).

☐ **YES! Upgrade me for $119 (reg. $149).**
☐ **No, please send me a FREE demo disk first.**

PowerBASIC, Inc.
316 Mid Valley Center,
Carmel, CA 93923
(408) 659-8000 — FAX (408) 659-8008
(800) 780-7707 — BBS (813) 625-1721

Offer expires 10/15/94.

PowerBASIC is a registered trademark of PowerBASIC, Inc. BASICA is a trademark of IBM Corporation.

Circle 111 on Reader Service Card

DOS RESOURCE GUIDE

Best of the Batch

Move 'Em Out

When I read Dan Gookin's column "Take a Chain Saw to Deadwood Directories" [*DRG*#10, July 1993, page 11], I thought, "Why not write a batch file that combines some of the functions of MS-DOS 6.x's XCOPY and DELTREE commands so that I can move files in one step?" The result is MOVEM.BAT (see the accompanying program listing), which grafts and prunes any part of your directory tree, including empty directories. The program includes safeguards to prevent you from accidentally deleting your root-directory files and all the subdirectories and files below it.

MOVEM.BAT moves files from any drive and directory to any other drive and directory, using the XCOPY command to copy the files. It then deletes the source files and their directories, using MS-DOS 6.0's DELTREE command.

The correct syntax for MOVEM.BAT is as follows:

```
MOVEM [source path]
   [destination path]
```

where *source path* is the path for the files or directories you want to move, and *destination path* is the new location for the files or directories. Typing the following DOS command, for example, moves to B:\OLDGAMES all the files in C:\GAMES, as well as all subdirectories and files below C:\GAMES in your directory structure:

```
MOVEM C:\GAMES B:\OLDGAMES
```

As written, the program uses similar wording to that in VCOPY.BAT, which appeared in "Unlocking the Secrets of Exit Codes" [*DRG*#7, January 1993, page 23]. Readers who have already typed in that listing may modify their existing VCOPY.BAT rather than typing in MOVEM.BAT from scratch.

To prevent you from losing important sections of your directory structure, the program makes good use of exit codes and error trapping, displaying detailed messages if something goes wrong. You needn't worry about catastrophic results. If you make a mistake, the worst thing that can happen is that you'll have to copy files back to their original locations. If the program doesn't finish its job, you may have to delete files manually from the destination drive or directory.

Brenton A. Textor
Folsom, Calif.

LET US HEAR
from you!

DOS Resource Guide wants to know about your computing trials and tribulations. The editors welcome your correspondence and encourage you to share your DOS tips or criticize ours.

From among the letters and batch files we receive, *DRG*'s editors will choose a "Best of the Batch" selection that could earn you $50 and a published credit.

So whether you work as a network administrator or high-school teacher, or just hang out at the command prompt, you have something to share with *DOS Resource Guide* readers. Send your criticism, your praise, and your best batch files to:

DOS Resource Guide
80 Elm St.
P.O. Box 802
Peterborough, NH 03458-0802

or reach us through CompuServe: 75300,2361
MCI Mail: 668-4855 or the *DRG* BBS: (603) 924-3181.

MOVEM.BAT takes advantage of MS-DOS 6.x's XCOPY and DELTREE commands, letting you graft files and directories to a new section of your directory tree and prune the old versions of the files and directories.

```
@ECHO OFF
REM NAME: MOVEM.BAT
REM PURPOSE: Moves files from one drive/directory
  to another.
REM SYNTAX: MOVEM [source path] [destination path]
REM EXAMPLE: MOVEM C:\GAMES B:\OLDGAMES
REM The above example moves to B:\OLDGAMES
  all files in
REM the directory C\:GAMES, including any
  of its subdirectories
REM and their files.
IF "%1" == "" GOTO SYNTAX
IF "%2" == "" GOTO SYNTAX
IF "%1" == "C:\" GOTO WARNING
IF "%1" == "c:\" GOTO WARNING
IF "%1" == "C:" GOTO WARNING
IF "%1" == "c:" GOTO WARNING
GOTO START
:SYNTAX
CLS
ECHO.
ECHO The correct syntax is: MOVEM [source path]
  [destination path]
ECHO.
ECHO Example: MOVEM C:\GAMES A:\GAMES
ECHO.
GOTO END
:WARNING
CLS
ECHO.
ECHO                            Warning!
ECHO.
ECHO The operation you requested would remove
  all files from the root
ECHO.
ECHO directory of hard disk C
ECHO.
ECHO to %2. This program can't perform
  that operation. If you simply
ECHO.
ECHO want to copy these files to another
  drive/directory, use COPY or
ECHO.
ECHO XCOPY.
ECHO.
GOTO END
:START
CLS
ECHO.
ECHO        Moving requested files from %1 to %2.
ECHO.
ECHO D|XCOPY %1 %2 /E /S /V
IF ERRORLEVEL 4 GOTO CATCH_ALL
IF ERRORLEVEL 2 GOTO CTRL_C
IF ERRORLEVEL 1 GOTO NO_FILES
IF ERRORLEVEL 0 GOTO DELETE
:DELETE
DELTREE /Y %1
GOTO SUCCESS
:SUCCESS
CLS
ECHO.
ECHO All files in %1 and its subdirectories
  have been moved to %2.
ECHO.
ECHO Directory %1 and all of its subdirectories
  have been deleted.
ECHO.
GOTO END
:CATCH_ALL
CLS
ECHO.
ECHO One of the following errors has occurred:
ECHO.
ECHO 1. There is insufficient memory to copy
  the requested files.
ECHO.
ECHO 2. There is insufficient disk space
  at the requested destination.
ECHO.
ECHO 3. You specified a drive or directory
  that doesn't exist.
ECHO.
ECHO 4. You mistyped the source/destination
  pathname.
GOTO TRY_AGAIN
:CTRL_C
CLS
ECHO.
ECHO The program was terminated because
  you pressed Ctrl-C.
GOTO TRY_AGAIN
:NO_FILES
CLS
ECHO.
ECHO The source directory, %1, doesn't exist
  or contains no files
ECHO.
ECHO to copy.
GOTO TRY_AGAIN
:TRY_AGAIN
ECHO.
ECHO Please correct the problem and try again.
ECHO.
GOTO END
:END
```

End

If you missed it on the newsstand,
here's your chance to order

MAXIMUM DOS

HIGH-POWERED TOOLS & TECHNIQUES
to tune your PC for MAXIMUM PERFORMANCE

EVERYTHING YOU NEED IS ALREADY IN DOS!

HOW TO:

- ★ Fine-tune AUTOEXEC.BAT and CONFIG.SYS to free up memory, load your favorite programs, and customize your PC setup.
- ★ Run DOS's own disk utilities to troubleshoot your hard drive and make it work faster.
- ★ Squeeze the last byte out of memory using MEMMAKER and EMM386.
- ★ Speed up hard drive reads and writes with SMARTDrive, DOS's top-notch caching utility.

THE MAXIMUM DOS TOOL KIT:

- Dozens of *DRG*'s best utilities reprinted and organized by function.
- Batch files and QBasic programs for taming disks and directories.
- Hardware Helpers: get control of the keyboard, cursor, and screen.
- Save typing time; all listings are on the disk!

DON'T MISS THE

COMPANION DISK

It's jam-packed with over 50 of our best batch files, Debug scripts, and QBasic programs, all designed to make you work faster and smarter. The same great programs from the magazine!

BONUS — Our experts' favorite shareware: 4DOS 5.0, a perennial best-seller that provides power and flexibility at the command line. HYPERDisk 4.65, a disk cache that speeds up your hard drive by placing frequently accessed data in RAM. LIST 9.0e, the classic replacement for DOS's TYPE command lets you display the contents of files easily.

FREE REFERENCE CARD

to DOS 6's built-in memory & disk utilities

YES! I WANT MAXIMUM DOS! PLEASE SEND ME :

Qty		DMX3	DMX5
	BOTH MAXIMUM DOS and the COMPANION DISK for $14.90	3.5″	5.25″
	Just the MAXIMUM DOS SPECIAL ISSUE for $5.95		
	Just the COMPANION DISK for $13.95	3.5″	5.25″

Please Circle Disk Size

Total Enclosed: ⇨

Canada, Mexico, and Foreign Surface add $5.50 per issue/disk ordered (includes applicable tax). Airmail add $9.50. Prepayment is required on all orders. U.S. funds on U.S. banks only. Magazines and disks may be shipped separately. Please allow 3-4 weeks for delivery.

PDRG/001/A17S4

☐ Check or money order enclosed (payable to DOS Resource Guide)
Charge my: ☐ VISA ☐ AmEx ☐ Discover ☐ MC
Card # .... Exp. Date ....
Signature ....
Name ....
Address ....
City .... State .... Zip ....

MAIL TO: DOS Resource Guide,
P.O. Box 672, Mt. Morris, Il 61054

Or Call:
1-800-349-7327

GOOKIN ON DOS

Cool DOS Utilities Still Exist

by Dan Gookin

The days of DOS utilities may be gone, but that doesn't mean enterprising programmers aren't developing new ways of dealing with the old gal. In fact, nothing tickles me more than stumbling on some newfangled way of solving an old DOS headache. One such suite of programs is 6th Sense, a bunch of nifty DOS 6 shareware utilities designed by Chuck Steenburgh of Tay-Jee Software.

Chuck got his inspiration from a DOS 6 review in which the writer praised the new DIR and FORMAT commands for doing a lot, but complained that setting all the various switches and options was a pain. Chuck sat down with Power C and powered out two nifty programs that let you easily set options for DIR and FORMAT and optimize them.

> *I've always felt that Microsoft made DOS lame on purpose, as a taunt to basement programmers to do something better.*

Setting the DIR Command's Options

The DIR command is possibly the most commonly typed DOS command. It also has a whole parade of switches marching after it. The options they control —various sorts and filters—could make it powerful if they weren't so cryptic. Enter 6th Sense's DCPLUS utility, DIR Commander Plus.

DCPLUS won't win any awards for fancy pull-down menus, bleeping boxes, or charming multimedia. Instead, it presents you with a straightforward list of all the various DIR command options and settings—stuff the manual takes a good three pages to describe and at which it fails miserably. (See the accompanying screen "DCPLUS's No-Nonsense Display.")

You set the options by pressing various keys: *P* to pause the display; *O, N* to sort files by name; *L* for lowercase; and so on. The screen fills with various Xs that tell you which of DIR's many options you've selected. To display a directory, press the F10 key, enter a wildcard option (or just press Enter to see all the files), and your customized DIR splashes on screen. If you want to make your options permanent, press *D* to select DIRCMD Set. It places into memory an environment variable (DIRCMD) that forces DIR to always run with your preset options.

One drawback is that DCPLUS doesn't place the option into your PC's AUTOEXEC.BAT file. Instead it creates a batch file, SETFILE.BAT, containing the proper DOS commands to configure DIR the way you specified. So to set things up in AUTOEXEC.BAT, all you need to do is "call" the SETFILE.BAT program. You can do this by placing the following line into your AUTOEXEC.BAT file:

```
CALL SETFILE.BAT
```

Then again, if you're fond of the way DCPLUS sets things up for you, add it to your set of start-up programs by including this line in AUTOEXEC.BAT instead:

```
DCPLUS
```

Just make sure that the DCPLUS file is on DOS's search path, and that you've already used the PATH command to set it up somewhere before the line above in your AUTOEXEC.BAT file.

The WarpFormat Secret

Even more useful than 6th Sense's DCPLUS utility is Chuck's formatting help program, FCDR. Longtime readers of this column will remember when I tried to

Take the Dread Out of DOS!

PLUS 13 DOS TIPS!
Dan Gookin Takes a Peek at MS-DOS 6.2
DOS RESOURCE GUIDE
The PC Productivity Magazine
THE DOS WARS HEAT UP
HOW PC DOS STACKS UP AGAINST MS-DOS
DoubleSpace Changes Some DOS Habits
Break the Speed Barrier in QBasic
How to Track Down DOS Help On-Line
10 REASONS WHY YOU CAN'T IGNORE DOSKEY
Hard Drive Swap Flop: Don't Let this Happen to You!
INSIDE: Shareware Picks, Reader
Dan Gookin: DOS 6.2 Miracle Debugger Saves Doomed Batch Files!
BRAVE NEW DOS
4 Reasons to Care About Novell DOS 7
9 DOS TIPS You Can Use Today!
FREE BATCH UTILITIES
Compress Data On the Fly
WOW! Electronic Index/Autodialer
Find a BACKUP Formula That Fits
New Column: Your Best QBasic Ideas
FREE Game Inside
Desperately Seeking a Searcher?
Flex Load Files Faster than DOS
COOL TIP: Hardin Brothers on Batch Files with Mouse Control

SUBSCRIBE TO DOS RESOURCE GUIDE:

Get 6 issues of DOS Resource Guide for $23.70.* (That's just $3.95 an issue!)

Save even more on12 issues: $44.00.* (That's 25% off the cover price!)

1-800-349-7327

DOS RESOURCE GUIDE**, the bimonthly periodical packed with tips, tutorials, and information on all aspects of PC computing, is your antidote for computing confusion.**

Beginners will find step-by-step help for copying files, creating and removing directories, and practical advice on using DOS's wildcards to cut down on keystrokes.

More advanced users will learn how to use batch files, and how to automate tedious activities such as backing up large groups of files. And you'll learn to use DOS's QBasic programming language. Each issue of *DOS RESOURCE GUIDE* is filled with articles about DOS fundamentals, tips on using DOS commands, undocumented shortcuts, and batch files you can type and run.

Start today! Tap all of your computing potential by learning DOS, personal computing's most important operating system. Use the coupon , or call (toll-free):

1-800-349-7327

SAVE UP TO 25%

SUBSCRIBE

☐ **YES, Send me 6 issues of *DOS Resource Guide* for $23.70. *Save me a buck an issue!**

☐ **YES, Save me even more: I'll take 12 issues of *DOS Resource Guide* for $44.00.* That saves me 25% off the cover price!**

Name: ____________________

Address: ____________________

City: ____________________

State/Zip: ____________________

☐ Payment enclosed ☐ Bill me

Charge my: ☐ MasterCard ☐ VISA ☐ AMEX ☐ Discover

Card #: ____________________

Expire Date: ____________________

Signature: ____________________

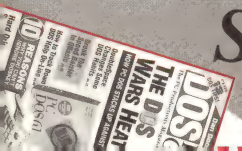
*In Canada and Mexico pay $29.70 (12 issues $55.00); all other foreign surface mail pay $41.70 (12 issues $78.00); foreign air mail pay $53.70 (12 issues $99.00). Foreign/Canadian orders must be prepaid in U.S. funds drawn on a U.S. bank. Please allow 6–8 weeks for delivery.

A17DRG

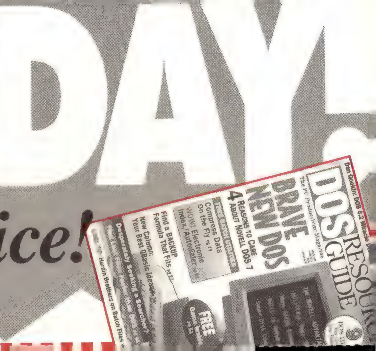
MAIL TO: DOS Resource Guide, P.O. Box 669, Mount Morris, IL 61054

SUBSCRIBE TODAY!

Save up to 25% Off the Cover Price!



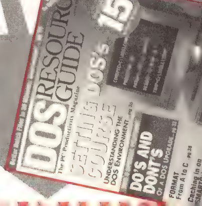
DOS
The PC Productivity Magazine
THE DOS WARS HEAT
HOW PC DOS STACKS UP AGAINST
DoubleSpace Changes Some DOS Habits
Break the Speed Barrier in QBasic
How to Track Down DOS Help On-Line
10 REASONS
PC DOS 6.1



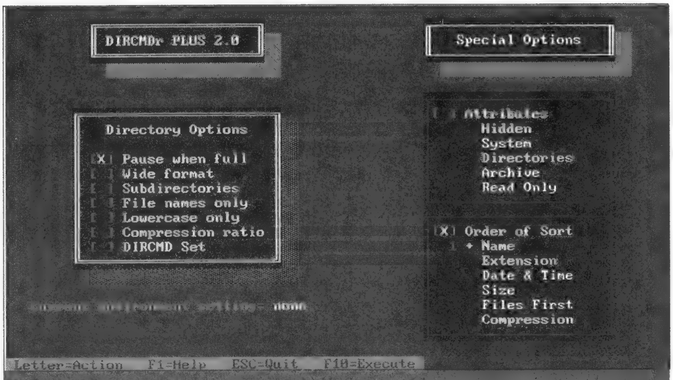
Dan Gookin: DOS 6.2 Miracle
DOS RESOURCE GUIDE
The PC Productivity Magazine
BRAVE NEW DOS
4 REASONS TO CARE ABOUT NOVELL DOS 7
Compress Data On the Fly pg 39
WOW! Electronic Index/Autodialer pg 62
Find a BACKUP Formula That Fits pg 22
New Column: Your Best QBasic Ideas
Desperately Seeking a Searcher?
Hardin Brothers on Batch Files
FREE Game Inside

YES! SAVE ME MONEY

12 ISSUES $44.00


DOS RESOURCE GUIDE
The PC Productivity Magazine
GETTING DOS's COURSE
UNDERSTANDING THE DOS ENVIRONMENT ...pg 26
15
DO'S AND DONT'S
OF A DOS UPGRADE...pg 32
FORMAT From A to C ...pg 38
Caching in on SMARTDrive...pg 57
ck-Tock
ck Your Clock
PROGRAM!

6 ISSUES $23.70



[X] Pause when full

[X] Order of Sort
1 ♦ Name

DCPLUS's No-Nonsense Display. It may not be beautiful, but DCPLUS's list of DIR command options gets right to the point.

write an all-encompassing disk-formatting batch file. That's all passé now, since the FCDR utility will set up various options for the FORMAT command.

FCDR is Format Commander. It contains a list of options and various settings the FORMAT command allows—including some specialties Chuck added himself, such as the ultra-fast "WarpFormat" option. To set these items, you use the handy FCDR interface (see the screen "Setting Format Options with FCDR"); you can then format a disk or use your options to create a FORMCMD environment variable.

Don't be confused; FORMCMD isn't an "official" DOS environment variable, not like DIRCMD. Instead, it's used by another of Chuck's programs, FMT.BAT. That batch file uses the FORMCMD environment variable to format your disks to the specifications configured in the FCDR utility.

The advantage of having a FMT.BAT program is that you can type FMT to format a disk instantly and never have to worry about setting any other options. And since it doesn't actually mess with the original FORMAT command, you can still use FORMAT to format disks if you want to.

Now for the secret: When you select WarpFormat from FCDR's main menu, you tell DOS to format a disk without any prompting. Three optional switches for DOS's FORMAT command are undocumented: SELECT, BACKUP, and AUTOTEST. The AUTOTEST switch is the one FCDR uses for its WarpFormat option. AUTOTEST tests the disk in the drive to determine its size, then formats the disk to that particular size without displaying any prompts or information. It can be a deadly option—but if you're writing a disk-formatting batch file or utility, it's invaluable.

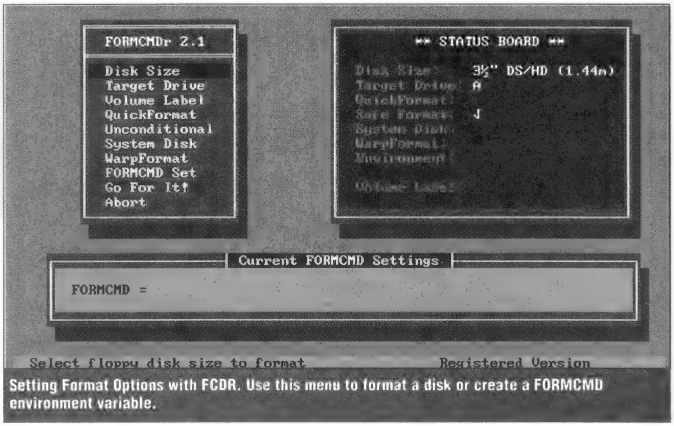
But Wait! There's More . . .

The 6th Sense package of utilities also comes with a long batch file called MACROS.BAT. It contains perhaps the best DOSKEY macros I've ever seen, including ones to find files and report free disk space. It's fully documented, so you can actually learn something about DOSKEY and your computer, too.

Discovering a utility like 6th Sense may not seem like a big deal in the era of Windows flash and dazzle. But stuff like this made using DOS a lot less headachy in the early days. I think it's charming that such utilities still exist, and lots of other DOS maniacs will enjoy such programs as well. I've always felt that Microsoft made DOS lame on purpose, as a kind of taunt to basement programmers everywhere to do something better. Utilities like 6th Sense prove that there's always a better way to deal with some of DOS's shortcomings, and I'm happy to share them with you.

6th Sense is available on CompuServe (or from *DRG*'s BBS: 603-924-3181) as shareware; register it if you make it a part of your PC's software repertoire. Also available on CompuServe is Steenburgh's Stuff, a collection of other DOS and batch-file utilities. For more information, contact Tay-Jee Software, P.O. Box 835, Lexington, VA 24450-0835; 800-378-3966. ■

Contributing Editor Dan Gookin is the author of numerous books on computing, including the best-selling DOS for Dummies *(IDG Books).*



Disk Size: 3½" DS/HD (1.44m)
Target Drive: A
QuickFormat:
Safe Format: J
System Disk:
WarpFormat:
Environment:
Volume Label:

FORMCMD =

Select floppy disk size to format

Setting Format Options with FCDR. Use this menu to format a disk or create a FORMCMD environment variable.

READER FORUM

A Wrench In the Works

by Hardin Brothers

Way back in the Dark Ages of personal computing—sometime in the early '80s—I built my first DOS machine, a Heath/Zenith 150, from a kit. That was the easy part. After I'd soldered each of the hundreds of components into place, tested every one of the boards, and run the computer's self-diagnostic software, it was time to boot up MS-DOS. I expected it to be easy to learn because, after all, I'd had extensive experience with various other operating systems.

No such luck. I still remember the frustration as I struggled for weeks to make sense of DOS—and the number of times I needed help. Those feelings came streaming back recently as I moved first into OS/2 as the prime operating system on one of my computers and then into Unix through a local Internet connection.

The only "dumb questions" are the ones that never get asked. Speak up—because working through the answers is the only sure way we learn.

I'm sure I've committed every mistake a beginner can make, asked every dumb question imaginable, and made all the wrong choices from every menu system I've seen. More than anything else, the experience has reinforced my belief that the only dumb questions are the ones that never get asked.

The questions I've received for "Reader Forum" this month are light-years ahead of those I've been asking recently. Lucky for me, they're about an operating system I know fairly well. If you have a question, don't be afraid to send it in c/o *DOS Resource Guide*. Asking questions and working through the answers is the only sure way we learn.

Visualizing QBasic

Edgar C. Slack of Somerset, New Jersey, presents an interesting QBasic problem: He's written a program that uses numerical data to draw graphs on screen and wonders whether he can print them from QBasic. The short answer is yes, but getting there may require a side trip through Windows.

It's easy to transmit simple text to a printer from a QBasic program, the DOS command line, or almost any application. The software just sends characters to the printer, one at a time, and the printer puts them on paper.

Printing graphics is harder. Are you using a dot-matrix printer or a laser model? If it's a dot-matrix, how many pins does it have, and how are they spaced? If it's a laser printer, is it compatible with PostScript or with PCL, Hewlett-Packard's Page Control Language? Which version of which language does it use? How many dots per inch do you want? Can your printer display color? Do you want the image in landscape or portrait mode? The questions go on and on.

Want to Know More?

If you have a question, or need further help with something that has appeared in DRG, write:

Reader Forum
DOS Resource Guide
80 Elm St.
P.O. Box 802
Peterborough, NH 03458-0802

Please include a copy of your AUTOEXEC.BAT and CONFIG.SYS files on disk if you have particular problems with software or hardware conflicts and compatibility.

Many DOS applications that print graphics or complex text (with underlines, superscripts, subscripts, multiple fonts, and so on) include support for several printer models. You tell the application which one you're using, and the program adjusts itself to your equipment. But when you write your own programs with QBasic or any other language, you're on your own.

If you're really ambitious, you can get out your printer manual, figure out how to send graphics images to it, and write a print routine into your QBasic program. You'll probably find that it takes several times as much effort to write the print section of the program than the display section.

A second possibility is to use the Print Screen key, but that will work only if you've loaded the DOS TSR (terminate-and-stay-resident) program called GRAPHICS.COM before you start the QBasic program, and only if your printer is one of the models supported by GRAPHICS.COM and its GRAPHICS.PRO data file. Instructions for GRAPHICS.COM are available from DOS's online help. (If GRAPHICS.COM doesn't support your particular printer model, its manufacturer may be able to suggest a comparable program.)

Windows offers a third, and much more powerful, option. One of Windows's advantages is that once you install the appropriate printer driver, all applications can send text or graphics to it with equal ease. If you're using Windows, run your QBasic program within it. Copy the graph to the Windows clipboard, then into a Windows paint program (such as Paintbrush, included with Windows) or a Windows word processor (such as Write, which comes with Windows, or Microsoft Word for Windows). You can then edit the graph, resize it, and print it easily.

Another option, if you're using Windows, is to rewrite your application in Visual Basic for Windows, which features built-in printer support. It's much easier to print graphics from a Visual Basic for Windows program than a QBasic program, simply because all the printer support you need is built into Windows and the drivers you've already installed.

Defeating Single-Step Bootup

Linda Dickert of Beaufort, South Carolina, has written a password-protection program for the network of 18 computers she manages at a local community college. She explains that when the multiple-configuration menu appears on screen, DOS displays the following instructions:

```
F5=Bypass startup files  F8=Confirm each line of
          CONFIG.SYS and AUTOEXEC.BAT
```

Linda would like to get rid of the the instructions or disable the F5 and F8 keys during bootup. The key here is DOS's SWITCHES command. SWITCHES isn't publicized as much as it should be; it's very helpful in system security.

The syntax of DOS's SWITCHES command—which you can use only in a CONFIG.SYS file—is the following:

```
SWITCHES= [/F] [/K] [/N] [/W]
```

You can use one or more of these options on the same line, depending on your needs (the /F and /N options are most appropriate for Linda's situation):

- SWITCHES=/F forces DOS to skip the two-second delay after it displays the "Starting MS-DOS . . ." message. It lets you boot your computer a little faster and get through CONFIG.SYS and AUTOEXEC.BAT processing faster than many users expect.
- The /N option disables the F5 and F8 keys during multiconfiguration processing. If you include the line SWITCHES=/N at the beginning of your CONFIG.SYS file, other users won't be able to get around your menus (unless they boot from a floppy disk, of course).
- For those interested, /K adds some compatibility for old programs that don't understand an enhanced keyboard; it turns your 101-key keyboard into an old-style 88-key keyboard. If you use this option and also load ANSI.SYS, you must add the /K switch to the DEVICE=ANSI.SYS line.
- Finally, the /W switch is just for users of Windows 3.0 on a 386 or later computer. It lets you move the Windows helper file WINA20.386 from your root directory to a directory of your choice. If you use the /W switch and move the file, you must also add a DEVICE command to the [386ENH] section of your SYSTEM.INI file to tell Windows where to find the file. If you're using Windows 3.1 or Windows for Workgroups, you should never use SWITCHES=/W.

Broken Batch Files

Thomas Dunnahoo of Azusa, California, offers a computing mystery. He writes that his batch file DELA.BAT contains the following line:

```
DEL *.* < C:\BAT\Y.TXT
```

The file Y.TXT contains a *Y* and a carriage return. Thomas reports that the batch file worked for months, but has now quit working. DEL asks its question, but when Thomas types Y, the computer locks up. Trying a suggestion he found in *DRG*#15 (the

May 1994 issue), he's replaced the broken line with the following one:

```
ECHO Y | DEL *.*
```

When DOS executes this line, however, it generates a piping error and an endless loop of "access denied" messages on Thomas's computer.

When something works "for months" and then stops working, there must have been a change in the hardware, the operating system, or associated software. In Thomas's case, both batch-file lines should work just fine, so he needs to look for the problem somewhere else.

Assuming that Thomas's other programs and batch files work okay on his computer, it doesn't sound like a hardware problem, but rather a change in DOS, in configuration files, or in memory-resident (TSR) utilities. Although I can't offer a definitive solution without checking out Thomas's computer in person, here are some suggestions for those of you who find yourselves in a similar situation.

First, make sure your hard disk contains some free space and that the root directory isn't full. If you can copy a file from a subdirectory to the root directory and then delete the file, your hard disk is okay.

Next, make sure your PATH isn't too long. The total permissible length is 127 characters, including the obligatory PATH=, which you see when you type SET. Any environment variable that's too long, including PATH, can cause all sorts of strange errors.

Also, see whether other piping and redirection commands will work from the command line and within simple batch files. You could try these three simple commands:

```
DIR > TEMP.FIL
MORE < TEMP.FIL
DIR | MORE
```

If they don't work, something is interfering with DOS's redirection routines. If they do work, something may be interfering with the DEL command.

The next step is the most difficult. You may have added a memory-resident program or a device driver that's causing a problem. First make a backup (preferably on a floppy disk) of your current CONFIG.SYS and AUTOEXEC.BAT files. Also, make sure you have an emergency bootup disk, which you can also use to edit the configuration files on your hard disk. Once you do, edit both CONFIG.SYS and AUTOEXEC.BAT, adding the word REM plus a space in front of each line that's not absolutely necessary.

You'll have to decide which lines to leave and which to exclude temporarily with the REM command. You may have to leave in your memory manager (HIMEM.SYS or its equivalent), any device drivers you need to access your hard drive (some disk-compression programs fall into this category), and anything you need to make your video screen work properly. Remove RAM drives, memory-resident utilities, PATH settings, SET commands, and everything else that's not absolutely essential. Now boot up and try the broken batch files again. If they work, then something in your CONFIG.SYS or AUTOEXEC.BAT file was getting in their way. Try removing some (not all) of the REMs and reboot. Do they still work? Keep adding or deleting REMs, rebooting, and testing until you can pinpoint the offending lines.

If you find that a memory-resident program is causing the problem, try changing the order in which you load TSRs—sometimes that will fix the problem. Or get in touch with the publisher of the program and see whether the company's tech-support staff can offer any help.

Finding the cause of a "mysterious problem" like this can be particularly time-consuming and frustrating. If you have any suggestions, write to me c/o *DOS Resource Guide*. I'll include some of the most promising solutions in future columns. ■

Technical Editor Hardin Brothers has been working with and writing about computers for 15 years.

Windows TIP

HOW MUCH RAM DOES A RAM HOG HOG?

If you use Windows 3.1, you may have discovered how to use the Control Panel's desktop icon to install a wallpaper (BMP) file as background for the Windows desktop. But you may not realize how much RAM a typical wallpaper image robs from your applications. For example, if you start Windows in 24-bit color mode, then load in a 640-by-480-pixel wallpaper background, Windows reserves a whopping 921K bit-map buffer, stealing almost a megabyte of RAM from your system. Screen savers—programs that display fanciful flying objects in place of a static screen—also gobble up memory.

To find out exactly how much RAM an *objet d'art* or screen saver hogs, pull down the Program Manager's Help menu, then select the About option. Jot down the total available RAM listed under System Resources. Switch back to the Control Panel, select the desktop icon, then deactivate your wallpaper image or screen saver. Return to the Program Manager, pull down the Help menu, and select the About option again. Subtract the figure you wrote down from the amount of RAM System Resources says is available when a screen saver or wallpaper isn't in use. The difference between the two figures represents the memory your wallpaper image or screen saver requires.

—John Wolfskill

UPDATE YOUR DISK DRIVES

Tame Your Fussy Floppy Drives

The DOS commands DRIVPARM and DRIVER.SYS can bridge the communication gap between you and those stubborn storage devices.

by Hardin Brothers

I've recently come across a series of questions about what seems to be the weakest and most frustrating part of any computer system: the floppy drives. I've always had problems with mine, and find that they're the least reliable parts of the systems with which I've worked. I've replaced dozens of them, made new cables, and even rescued disks that refused to turn. But these common questions stumped me at first:

- How can I fix DOS when it insists on displaying a directory from a disk I've removed from the drive?
- If I use only double-density (360K and 720K) disks, how can I stop DOS's annoying habit of trying to format them for 1.2MB and 1.44MB?
- How can I use a 3.5-inch drive or a high-density drive if my BIOS (basic input/output system) knows nothing about it?
- How can I add an external floppy drive to my computer?
- How can I copy from one 3.5-inch disk to another on a system with both a 5.25- and a 3.5-inch drive?

After I did some digging in my reference library, I unearthed two commands that will solve all these problems, yet I've seldom used them: DRIVPARM and DRIVER.SYS. (The latter is technically a device driver, not a command.)

The DRIVPARM command supplies the solution to the first three questions in my list; DRIVER.SYS takes care of the remainder. If you find yourself arguing with DOS about your floppy drives, these commands might just help you settle your differences. Both DRIVPARM and DRIVER.SYS can be used only in your CONFIG.SYS file. (Their location within it doesn't matter.)

Now if you're not in the habit of editing CONFIG.SYS and testing new commands, be sure to read the sidebar "Safety First" so that you can proceed without endangering any of your computer's components.

Setting Drive Parameters

Normally, your computer's setup program holds information about your floppy disk drives and their capacities. But your drive may not fit a standard BIOS definition, or it may have special requirements. That's when you should call on DRIVPARM to establish better communication between the drive and DOS.

You can use DRIVPARM only if your BIOS and DOS already recognize a drive, but, for some reason, are unable to work with it to your satisfaction.

Unfortunately, DRIVPARM's syntax can be intimidating, which is why many users seldom even consider trying it. (See the accompanying sidebar "DRIVPARM Syntax" for a complete description.)

Most floppy drives can send a signal to the computer when you open the door. DOS uses this signal to decide whether it should reread a disk's directory and file-allocation table (FAT). If DOS thinks a disk drive can send such a signal, it usually won't reread these vital system areas until you change disks. But if it thinks the drive can't send the door-open or "change-line" signal, it will reread the directory and the FAT each time it begins a new drive access.

Technical Editor Hardin Brothers has been working with and writing about computers for 15 years.

One of the most frequent uses of DRIVPARM is to set DOS straight about a drive's changeline capabilities. I've seen the signal break after working successfully for a year or two, either because of a crimp in the drive cable inside the computer or because of a failed sensor in the drive itself. In all cases, you can still use the drive, although not quite as efficiently, by telling DOS that it shouldn't expect a changeline signal. The command for a 1.2MB drive A is:

```
DRIVPARM=/D:0 /F:1
```

Notice that I've omitted changeline information (the /C parameter) to tell DOS that it shouldn't rely on a changeline at all. If the drive has a working changeline signal, the comparable command is:

```
DRIVPARM=/D:0 /F:1 /C
```

Avoiding Format Foibles

If you care about your data, you will never try to format a double-density 360K or 720K disk to 1.2MB or 1.44MB. But DOS has an annoying habit of trying to format floppy disks to the highest capacity of the drive. You can override DOS with FORMAT's various switches, but if you forget, you'll ruin a double-density disk (at least until you have a chance to run it by a bulk eraser).

If you always use double-density disks (or high-density disks on a 2.88MB drive), you can use DRIVPARM to set a new maximum capacity for a drive. For example:

```
DRIVPARM=/D:0 /F:0 /C
```

tells DOS that drive A (which you select with the /D:0 switch) should be treated as a double-density drive (/F:0) only. On the other hand, the following command:

```
DRIVPARM=/D:1 /F:2 /C
```

tells DOS that drive B has a maximum capacity of 720K. Both lines tell DOS to expect changeline signals from the drive.

SAFETY FIRST

To try the ideas and commands here, you will have to make some minor modifications to your CONFIG.SYS file. Because a typing error or other mistake in CONFIG.SYS can prevent a computer from booting, you should have an emergency boot disk handy. If you don't have one, put a fresh disk into drive A and type this line:

```
FORMAT A: /S
```

Then try to boot from the floppy and make sure you can edit the CONFIG.SYS file on your hard disk. You can store other files on your emergency boot floppy, including backup copies of your hard disk's CONFIG.SYS and AUTOEXEC.BAT files, utility programs that will help you recover from some hard-disk crashes, and so on.

But at the very least, you should have a floppy disk from which you can boot up and that you can then use to edit the configuration files on your hard disk.

If your computer is running under DOS 6.x, you can reduce the chance of a CONFIG.SYS hang-up by using multiple configuration menus. Create a new menu option in your CONFIG.SYS file with which you can experiment. If you make a mistake, you can reboot and select one of the options you're sure will work, then edit the offending line in the experimental section of CONFIG.SYS. You can also press F5 or F8 at boot time to skip all CONFIG.SYS processing or to process just one line at a time and skip the offending entry.

—H.B.

You'll see two interesting results if you set a drive to less than its maximum capacity. First, it may have an easier time reading a double-density disk. I've managed to recover data from a seemingly destroyed disk with this trick (but don't expect it to always work). You'll still be able to read from and write to high-density disks in your drives, though. Second, you can format the drive only at the capacity you've set or less. So you'll be able to use the FORMAT command without any switches to format double-density disks in those drives.

And, with DOS's FORMAT command and most similar utilities, you won't be able to format high-density disks in those drives at all.

Is this an advantage? If you stick to double-density disks for all your work, it is. If you switch between double- and high-density disks, you can use the DRIVER.SYS command described below (see the section "Adding More Drives") to make one drive appear as both a double-density and a high-density drive.

Upgrading an Old Computer

Finally, you can use DRIVPARM if you want to install a floppy disk your BIOS doesn't recognize. For example, many older computers know nothing about 3.5-inch drives. You could upgrade your BIOS, of course, to install a 3.5-inch drive, or you could use DRIVPARM instead.

The /I switch forces DRIVPARM to install the driver software for the drive you specify. For example, if your computer's BIOS doesn't recognize that drive B is a 1.44MB 3.5-inch drive, you could use this:

```
DRIVPARM=/D:1 /F:7 /I
```

With that line in your CONFIG.SYS file, you should be able to use your new 3.5-inch drive on any DOS computer as long as you use the correct cable and the drive is electronically compatible with your disk controller. In most cases, the /I switch will let you avoid purchasing a BIOS upgrade, which may not even be available for some older PCs.

Although you can use DRIVPARM to force DOS to set new parameters for a hard drive, tape drive, or optical drive, few people will use it for that purpose. Tape and optical drives are almost always shipped with their own driver software. Once you boot from a hard drive, you won't want to change that device's parameters. But you may be able to use DRIVPARM on an older computer to force DOS to recognize a large, second hard drive that can't be defined with the BIOS. That's one application of DRIVPARM I haven't needed, so you'll have to experiment to see whether it will work on your particular system.

Adding More Drives

We've seen that DRIVPARM changes the parameters of a drive that has already been recognized by DOS and your BIOS. But if you want to add a new drive, one that DOS wouldn't normally recognize at all, you need to use DRIVER.SYS.

The syntax for DRIVER.SYS is similar to DRIVPARM's, but more limited. (See the accompanying sidebar "Using DRIVER.SYS.") Don't dismiss DRIVER.SYS simply for that reason, though. It has two important uses that DRIVPARM can't duplicate.

Many disk controllers include provisions for four floppy drives instead of the two that are usually installed in the computer. The third and fourth drives are normally attached externally, in their own cabinet and with their own power supply. They're then connected to the computer with a standard floppy-drive cable. The problem is that your BIOS and DOS probably won't recognize the external drives once they're installed correctly. But DRIVER.SYS will come to your rescue and make the external drives visible.

DRIVPARM SYNTAX

The syntax of the DRIVPARM command (which you can use only in your CONFIG.SYS file) can be intimidating. Using Microsoft's syntax conventions, it looks like this:

```
DRIVPARM=/D:d [/F:n] [/H:h] [/T:t] [/S:s]
  [/N] [/C] [/I]
```

Of all these commands and switches, only the word *DRIVPARM*, the equal sign, and /D:*d* are required. /D:*d* names the drive for which you want to define new parameters. In DRIVPARM's method of numbering the drives, you must use /D:0 to represent drive A, /D:1 to represent drive B, /D:2 to represent drive C, and so on. Everything else in the line is optional, although you'll probably need to include at least one or two more pieces of information each time you use DRIVPARM.

For instance, you will almost always have to use DRIVPARM's /F:*n* option. /F:*n* specifies a disk "form factor," or drive type. DOS uses the following numbers to represent each drive type it supports:

- /F:0 means a 160K or 180K single-sided drive or a 320K or 360K double-sided 5.25-inch drive
- /F:1 means a 1.2MB 5.25-inch drive
- /F:2 means a 720K 3.5-inch drive
- /F:5 means a hard disk
- /F:6 means a tape drive
- /F:7 means a 1.44MB 3.5-inch drive
- /F:8 means a read/write optical disk
- /F:9 means a 2.88MB 3.5-inch drive

(If you're wondering what happened to /F:3 and /F:4, they were used originally to support 8-inch drives, a species now nearly extinct.) If you don't specify a /F: drive type, DOS assumes you're giving parameters for a 720K 3.5-inch drive (type 2).

If you're working with a standard floppy drive, you can skip the /H:*h*, /T:*t*, and /S:*s* parameters. These define the number of heads, tracks, and sectors the drive uses and are set automatically from the /F:*n* option. You'll need to set these options if you're using DRIVPARM to define a tape drive or optical disk drive. You may also need to set these three values if you're using an unusual floppy disk drive, such as one of those that can store 10 to 20 megabytes on a single disk.

The /N parameter tells DOS that the recording medium can't be removed. This parameter is generally reserved for hard drives, which are rarely set with DRIVPARM. You should never use DRIVPARM to define new parameters for your boot drive; if you do, you'll almost certainly lock up your machine, and you could damage the drive as well.

The presence or absence of the /C, or *changeline,* option can solve many problems with a faltering drive. If you use /C, DOS assumes that the drive will tell it when the door has been opened and closed. You can tell DOS that there is no changeline signal by specifying a DRIVPARM for the drive and omitting the /C option.

For example, if the changeline on your 1.2MB drive A is no longer working, you could add this command to your CONFIG.SYS file:

```
DRIVPARM=/D:0 /F:1
```

Finally, the /I option tells DOS that it should install a software driver for the drive in addition to setting its parameters. Use this option if you've added a 3.5-inch drive to a computer with an old BIOS that doesn't support these drives. This includes most 8088 computers and some early 80286 machines. The /I switch is an alternative to trying to find a BIOS update for your computer just to use a 3.5-inch disk.

—H.B.

USING DRIVER.SYS

The format syntax for DRIVER.SYS (which you can install only from CONFIG.SYS) is the following:

```
DEVICE=DRIVER.SYS /D:d [/F:n] [/H:h] [/T:t] [/S:s] [/C]
```

If you prefer, you may use DEVICEHIGH instead of DEVICE. Unless you've stored DRIVER.SYS in the root directory of your boot drive, you'll need to include its full path in the command (for example, DEVICE=C:\DOS\DRIVER.SYS).

The options for DRIVER.SYS look very similar, but they're somewhat less complicated than those DRIVPARM uses. DRIVER.SYS requires a drive number; lets you specify a drive by type or by heads, tracks, and sectors; and lets you specify whether or not the drive can report a possible disk change.

DRIVER.SYS supports only floppy drives, however, so both the /D: and /F: parameters work a little differently. DRIVER.SYS requires a drive number, specified with the /D:*d* parameter. Like DRIVPARM, DRIVER.SYS uses /D:0 to refer to drive A and /D:1 to refer to drive B. Unlike DRIVPARM, DRIVER.SYS uses /D:2 to refer to the third disk drive, which must be the first external drive, and /D:4 to refer to the second external drive.

Like DRIVPARM, DRIVER.SYS needs to know what drive type or form factor to use for the drive. /F:*n* uses the same numbering system as the DRIVPARM /F:*n* option, but DRIVER.SYS supports drives of types 0, 1, 2, 7, and 9 only:

- /F:0 means a 160K or 180K single-sided or a 320K or 360K double-sided 5.25-inch drive
- /F:1 means a 1.2MB 5.25-inch drive
- /F:2 means a 720K 3.5-inch drive
- /F:7 means a 1.44MB 3.5-inch drive
- /F:9 means a 2.88MB 3.5-inch drive

You can use the /H:*h*, /T:*t*, and /S:*s* options to set the number of heads, tracks, and sectors if you're using an unusual drive.

Finally, the /C option works the same way as its DRIVPARM counterpart. If the drive can send an open-door or changeline signal to DOS, you should include it. If the drive can't send that signal, omit the /C parameter.

—H.B.

The syntax for the third disk drive (a 1.44MB 3.5-inch floppy that supports the changeline signal) is this:

```
DEVICE=C:\DOS\DRIVER.SYS
  /D:2 /F:7 /C
```

The syntax for the fourth drive is the same, but you'd use /D:3 to name it. If you prefer, start the line with DEVICEHIGH= to move DRIVER.SYS into an upper memory block.

DOS will assign the floppy drives the next available drive letters in your system. For example, if you have one hard drive (and you don't use a disk-compression program such as DBLSPACE), the first external floppy drive will appear as drive D.

Phantom Drives

External floppy drives used to be common, but most of us get along just fine with one or two internal floppies and a large hard drive. DRIVER.SYS would probably have slipped into obscurity if it didn't have one other important use: It can assign a new drive letter to an existing floppy drive.

That may sound strange. Why would anyone want a floppy drive called both B and D? By using two letters for one drive, you perform some maneuvers that are normally impossible.

Assume for a second that you have both a 5.25-inch drive and a 3.5-inch drive in the same computer. How can you copy files from one 3.5-inch disk to another?

One way is to copy the files to your hard drive, swap disks, and then copy them to the new disk. But maybe you don't have sufficient hard-disk space for the files, or perhaps you have to copy files from several 720K disks to a smaller number of 1.44MB disks for archival purposes.

One of the cleverest ways to copy the files, with a DOS command such as COPY, XCOPY, or MOVE, is to give the drive two letters, perhaps

DOS TIP

TOO MANY COOKS SPOIL THE CACHE

If you use the MS-DOS disk-caching program SMARTDRV.EXE (SmartDrive), check the setting of the BUFFERS statement in your CONFIG.SYS file. When you use SmartDrive, it's best to set the value of the BUFFERS statement to less than the 20 to 30 buffers recommended for those who use a third-party disk-caching program. DOS 6.x's Setup program handles this automatically, changing BUFFERS to 10.

Besides saving valuable conventional memory (each buffer consumes 512 bytes of RAM), using a smaller value may improve performance. That's because the BUFFERS statement serves as a secondary disk-caching scheme. Setting the number of buffers too high may cause excessive data shuffling, which slows down hard-drive operations.

—John Wolfskill

B and D. In this case, the letter *B* refers to the normal second drive, and *D* is a "phantom" drive that uses the same piece of hardware.

COPY Made Easy

With a phantom-drive setup, you can use a DOS command such as the following:

```
COPY B:\*.* D:
```

If both B: and D: refer to the same drive, the computer will tell you when to insert the source and destination disks. (If you've copied files from A to B on a computer with a single drive, you've seen similar prompts.)

Another reason for creating a phantom drive is that you can use one letter for double-density disks and the other for high-density disks. Then you can format disks in that drive (or read some damaged disks) by using one drive letter for each density.

To create a phantom drive, just use DRIVER.SYS to assign a new letter to an existing drive. For example, to create a phantom drive letter for your second physical drive (drive B), and to set up the phantom drive as a 720K 3.5-inch drive with changeline support, you'd use this command:

```
DEVICE=C:\DOS\DRIVER.SYS
  /D:1 /F:2 /C
```

If you often copy data from one disk to another, you may find that phantom drives make your work much easier.

Listen Up

Using both DRIVPARM and DRIVER.SYS can help you work more efficiently with your disk drives. True, they won't clean the heads for you or make new cables, but they'll add a great deal of flexibility to the drives you already have—and you'll find them invaluable when your drives refuse to communicate with DOS the way you wish they would. ■

DOS TIP

SYSTEMATIC TRASH REMOVAL

I've developed a strategy that balances the need to prevent hard-disk clutter against the need to avoid deleting backup files just created by my word-processing program. At the heart of my system is TRASHCAN.BAT, a batch file that stores all BAK files in a directory called \TRASHCAN, keeps track of how old each file is, and purges files when they reach a certain age. At the end of each day, I back up all my data files and run TRASHCAN.BAT to search likely directories for BAK files and move them into \TRASHCAN. The sample batch file below, for example, searches my word-processing directory as well as the root directory.

TRASHCAN.BAT monitors the ages of the BAK files by changing their file extensions. When the program moves a file into \TRASHCAN, it changes its extension from BAK to 001. The next day, all files with the extension 001 are given the extension 002, and the BAK files are marked with 001 extensions. Each day, the files extensions are incremented by one. When TRASHCAN.BAT encounters files with the extension 010, it deletes them. Of course, you can modify the program to begin deletion earlier or later than that.

—Ronald L. Vanover

TRASHCAN.BAT keeps track of the age of your BAK files, purging them from your hard disk when they're too old to be useful.

```
@ECHO OFF
CLS
IF NOT EXIST C:\TRASHCAN\*.010
  GOTO SKIPTO9
DEL C:\TRASHCAN\*.010
REN C:\TRASHCAN\*.009 *.010
:SKIPTO9
IF NOT EXIST C:\TRASHCAN\*.008
  GOTO SKIPTO8
REN C:\TRASHCAN\*.008 *.009
:SKIPTO8
IF NOT EXIST C:\TRASHCAN\*.007
  GOTO SKIPTO7
REN C:\TRASHCAN\*.007 *.008
:SKIPTO7
IF NOT EXIST C:\TRASHCAN\*.006
  GOTO SKIPTO6
REN C:\TRASHCAN\*.006 *.007
:SKIPTO6
IF NOT EXIST C:\TRASHCAN\*.005
  GOTO SKIPTO5
REN C:\TRASHCAN\*.005 *.006
:SKIPTO5
IF NOT EXIST C:\TRASHCAN\*.004
  GOTO SKIPTO4
REN C:\TRASHCAN\*.004 *.005
:SKIPTO4
IF NOT EXIST C:\TRASHCAN\*.003
  GOTO SKIPTO3
REN C:\TRASHCAN\*.003 *.004
:SKIPTO3
IF NOT EXIST C:\TRASHCAN\*.002
  GOTO SKIPTO2
REN C:\TRASHCAN\*.002 *.003
:SKIPTO2
IF NOT EXIST C:\TRASHCAN\*.001
  GOTO SKIPTO1
REN C:\TRASHCAN\*.001 *.002
:SKIPTO1
IF NOT EXIST C:\TRASHCAN\*.BAK
  GOTO WP
REN C:\TRASHCAN\*.BAK *.001
:WP
IF NOT EXIST C:\WP\MEMOS\*BAK
  GOTO ROOTDIR
REM If you have DOS 6.x, you
REM may replace the COPY and
REM DEL commands below with
REM this command:
REM MOVE C:\WP\MEMOS\*.BAK
  C:\TRASHCAN
COPY C:\WP\MEMOS\*.BAK
  C:\TRASHCAN
DEL C:\WP\MEMOS\*.BAK
:ROOTDIR
IF NOT EXIST C:\*.BAK GOTO
  BANNER
REM If you have DOS 6.x, you
REM also may use MOVE here:
COPY C:\*.BAK C:\TRASHCAN
DEL C:\*.BAK
:BANNER
CLS
END
```

End

DOS & HARDWARE

Solving IRQsome Problems

Add-ons sometimes don't work because they need a hardware hook that another device is already using. MS-DOS's Diagnostics program can solve the problem.

by Ken Johnson

You've just installed a new sound card. When you boot your PC, however, you find that your modem doesn't work. The lights flash, but you can't dial out. And sometimes when you try to use the modem, your computer freezes, and you have to reboot. What's happening?

Welcome to the wonderful world of interrupt-request (IRQ) conflicts. Your sound card and modem are trying to talk to your computer on the same line. They're like two reporters fighting for one phone—only one is going to get the story to the main desk.

In this article, I'll begin by looking at IRQs and what might be causing your modem problem. Then I'll talk about Microsoft Diagnostics, a program that can help you research your system and head off IRQ problems.

Ken Johnson is a training and support manager at the law firm of Mayer, Brown & Platt in Chicago. He writes often about computers for PC users' groups and legal computing periodicals and is a contributing editor for Hard-Copy, *the journal of the Chicago Computer Society.*

May I Have Your Attention?

An IRQ is a request for the CPU's (central processing unit) attention—sort of an "Interrupt what you're doing and service me" message. Devices such as the keyboard and disk drives, and peripherals such as printers that attach to a parallel or serial port, are constantly vying for the CPU's consideration through IRQs.

Each interrupt request—identified by a number—is passed to the CPU by two interrupt controllers. IRQs are then handled in order of importance. For example, when you type, IRQ 1 tells the CPU that something urgent has occurred at the keyboard and needs to be processed.

Standard IRQ assignments are given to standard devices. (See the accompanying sidebar "Common IRQ Uses," opposite.) The first serial port (COM1) uses IRQ 4, and the second one (COM2) employs IRQ 3. The first parallel port (LPT1) uses IRQ 7, and the second one (LPT2) uses IRQ 5.

The keyboard uses IRQ 1; the hard-disk controller, IRQ 14; and the floppy-disk controller, IRQ 6.

You can have only one active device per IRQ. Two devices trying to use the same IRQ is trouble in the making.

One device might get locked out, for example, and not be able to request the CPU's attention because the other device "owns" the IRQ. Even worse, some IRQ conflicts can cause irregular and nonreproducible system crashes.

You need to verify which IRQs are in use before adding any device that needs an IRQ; you can add that device if it can be configured to employ an unused IRQ.

One on One

This brings us back to the original problem outlined above: Because the modem did work before you installed that new sound card, you should suspect an IRQ conflict. The sound card's documentation may tell you which IRQ the manufacturer set it to use.

For example, IRQ 3 might be the factory default. But IRQ 3 is assigned to the second serial port, COM2. And what's on COM2? The modem. So a conflict over IRQ 3 is the problem.

To find a solution, you might be able to move the modem to COM1, but let's assume you have your mouse there. Luckily, many PC devices have jumper switches that let you shift to a different IRQ.

Now the question is: Which IRQs are unused and available for the sound card? You could guess (and maybe something else won't work), but there's a better way.

Microsoft Diagnostics

MS-DOS 6 includes a program that can help—Microsoft Diagnostics, or MSD. (A version of MSD is also included with Windows version 3.1.)

Actually, "diagnostics" is a misnomer here. MSD doesn't perform diagnostic testing (for example, it won't point out your IRQ conflict); MSD is more of a system-analysis and reporting tool. With MSD's information, you can start diagnosing problems on your own.

Run MSD by typing `MSD` on the DOS command line. Note that you should run MSD directly from DOS and not under Windows. If you try to execute MSD under Windows, you'll receive a warning that some information won't be accurate. This is because Windows controls certain system resources (such as extended memory) that MSD just can't detect.

As MSD loads, you should see a message that tells you MSD is examining your system. If MSD fails to load or hangs up while doing the examination, start it again with the /I switch (MSD /I) to skip the initial hardware detection. MSD then inspects different parts of the system as you request the information.

On the Menu

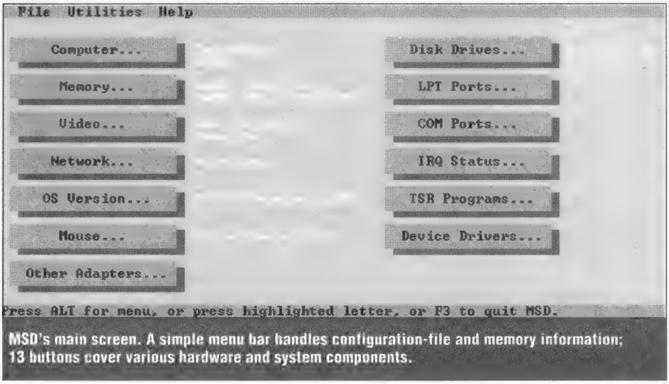
Once MSD is loaded, you'll see a menu bar at the top of the screen, along with 13 buttons for viewing information on the different elements of your PC. (See the accompanying screen showing MSD's main interface.)

Choose a button by pressing the highlighted letter displayed on the button or by clicking on the button with your mouse. To choose an item on the menu bar (File, Utilities, or Help), press the Alt key plus the first letter of that item (for example, Alt-F for the File menu). Or you can click on the item with your mouse.

You can display most of MSD's information by choosing among its buttons. For instance, to see your current IRQ assignments, select the IRQ Status button.

If there's more information than can be displayed on one screen, use the Page Up and Page Down keys to scroll through the details. If you have a mouse, you can use the scroll bars on the right of the display screen.

To return to the main screen, just press the Enter key or select the OK button if you prefer.



```
Press ALT for menu, or press highlighted letter, or F3 to quit MSD.
```

MSD's main screen. A simple menu bar handles configuration-file and memory information; 13 buttons cover various hardware and system components.

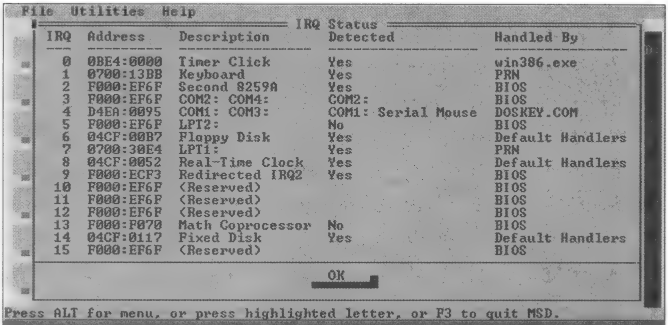
COMMON IRQ USES

Following are the most common IRQ uses on 286 and above computers:

IRQ 0	timer
IRQ 1	keyboard
IRQ 2	cascaded from second interrupt controller (IRQ 8–15); any board set to IRQ 2 will be redirected automatically to IRQ 9
IRQ 3	serial port 2 (COM2 and COM4)
IRQ 4	serial port 1 (COM1 and COM3)
IRQ 5	parallel port 2 (LPT2)
IRQ 6	floppy disk
IRQ 7	parallel port 1 (LPT1)
IRQ 8	real-time clock
IRQ 9	redirected from IRQ 2 (appears as IRQ 2)
IRQ 10	reserved for future expansion (open)
IRQ 11	reserved for future expansion (open)
IRQ 12	reserved for future expansion (open); mouse port on PS/2 computers
IRQ 13	math coprocessor
IRQ 14	hard disk
IRQ 15	reserved for future expansion (open)

Back to the Sound Card

To find an open IRQ for the now-infamous sound card, run MSD and choose the IRQ Status button. You'll see which IRQs are in use (detected is indicated by a Yes), and which are free. In the hypothetical case described above, both COM2 and the modem are using IRQ 3.



```
IRQ  Address     Description          Detected              Handled By
---  ---------   -------------------  --------------------  ----------------
  0  0BE4:0000   Timer Click          Yes                   win386.exe
  1  0700:13BB   Keyboard             Yes                   PRN
  2  F000:EF6F   Second 8259A         Yes                   BIOS
  3  F000:EF6F   COM2: COM4:          COM2:                 BIOS
  4  D4EA:0095   COM1: COM3:          COM1: Serial Mouse    DOSKEY.COM
  5  F000:EF6F   LPT2:                No                    BIOS
  6  04CF:00B7   Floppy Disk          Yes                   Default Handlers
  7  0700:30E4   LPT1:                Yes                   PRN
  8  04CF:0052   Real-Time Clock      Yes                   Default Handlers
  9  F000:ECF3   Redirected IRQ2      Yes                   BIOS
 10  F000:EF6F   (Reserved)                                 BIOS
 11  F000:EF6F   (Reserved)                                 BIOS
 12  F000:EF6F   (Reserved)                                 BIOS
 13  F000:F070   Math Coprocessor     No                    BIOS
 14  04CF:0117   Fixed Disk           Yes                   Default Handlers
 15  F000:EF6F   (Reserved)                                 BIOS

Press ALT for menu, or press highlighted letter, or F3 to quit MSD.
```

Current IRQ assignments. Clicking on the IRQ Status button opens a screen listing your system's IRQ availability. When you add a device, use this option to determine which IRQs are free.

But let's say you find IRQ 5 open. (See the screen, right, showing current IRQ assignments.) The second parallel port (LPT2) uses IRQ 5, and since most PCs have only one parallel port, you'll discover that IRQ 5 is often free and available.

The solution, then, is to set the jumper switches on the sound card to use IRQ 5 and reinstall the card. The card's documentation should provide instructions.

Put It in Writing

You can save time (and frustration) by knowing which IRQs are in use before you install a device. So it can be useful to keep a printout of IRQ status.

As a matter of fact, you'll probably want a complete system information report in case, for example, your system's battery fails and you have to reset the CMOS settings for your hard drive. You'll be able to find the correct values in the Disk Drives section of the MSD report. (See the sidebar "An Overview of MSD," on the next page, for details.)

You can produce a system information and configuration report in one of two ways.

If you run MSD interactively, go to the File menu and select Print Report. Here you can choose which information to include—you'll probably want all of it—and whether to print the report to your printer or to a file.

When the report runs, the program asks you for certain identification information first, including name, company, address, and comments, which MSD prints at the top of the report.

You can also have MSD produce the report and write it to a file directly from the DOS command line if you prefer.

If you run MSD with the /F switch (with the filename specified), it will prompt you for customer information, then write a complete report to that file.

Running MSD with the /P switch and filename will write the complete report to the specified file without asking you for customer information. (See the accompanying sidebar "MSD Command Syntax," below.)

Knowledge Is Power

Handling IRQ conflicts can be a problem because they occur behind

MSD COMMAND SYNTAX

Here's a brief look at the MS-DOS 6's Microsoft Diagnostics system-analysis command and its optional switches:

MSD [/I] [/B] [/F *filename*] [/P *filename*] [/S *filename*]

If you type `MSD` by itself, the program will run interactively, and you can review the technical information via its detailed screens.

/I bypasses MSD's initial hardware detection. Use this switch if you're having trouble starting MSD or if MSD isn't running properly. You'll also need this switch if you're running MSD under Windows 3.0 in 386 Enhanced mode.

/B runs MSD in black-and-white instead of color.

/F *filename* writes a complete MSD report to the file specified (include the full path if you don't want the file in the current directory). The program will prompt you for identification information, including name, company, address, phone number, and comments, which will appear at the beginning of the file.

/P *filename* writes a complete MSD report to the file specified, without prompting you for customer information.

/S *filename* writes a summary MSD report (the information shown on the main MSD screen). If you include a filename, MSD will write the report to that file. If not, it will write the summary report to the screen. In either case, MSD won't prompt you for identification information.

—K.J.

the scenes in your PC and may not be immediately recognizable. And as you update and upgrade your system with new devices such as modems, a BUS mouse, sound cards, scanners, and CD-ROM drives, the probability of IRQ conflicts increases.

Many of these peripheral devices may require that you set them to a free IRQ, so you have to know which IRQs are currently in use and which are still available. But rather than guess, use DOS 6's handy Microsoft Diagnostics program to find out for sure. ■

AN OVERVIEW OF MSD

Here's a description of the information MSD provides through the program's main screen buttons:

Computer. Displays the PC and RAM BIOS manufacturer, processor type, keyboard, DMA controller configuration, and math coprocessor status.

Memory. Displays a map of the upper memory area between 640KB and 1MB.

Video. Shows the video-card manufacturer, model, and type; video BIOS version and date; and current video mode.

Network. Displays the network-specific configuration if your PC is connected to a network.

Operating System. Shows the DOS version, location of DOS in memory, boot drive, environmental settings, and path to MSD.

Mouse. Lists the mouse-driver version, mouse type, mouse IRQ number, and other mouse configuration information.

Other Adapters. Displays the game-card status for up to two game devices or joysticks.

Disk Drives. Displays sizes and number of free bytes for local and remote drives.

LPT Ports. Shows the port addresses of all installed parallel ports and the status of each port.

COM Ports. Lists the port addresses and communication parameters of all installed serial ports, as well as the status of each port.

IRQ Status. Displays the assignments and configuration of current IRQs.

TSR Programs. Shows the names, sizes, and locations of programs loaded in memory at the time you run MSD.

Device Drivers. Lists the names of all device drivers installed at the time you run MSD.

The MSD menu bar provides additional tools for viewing system information, as follows:

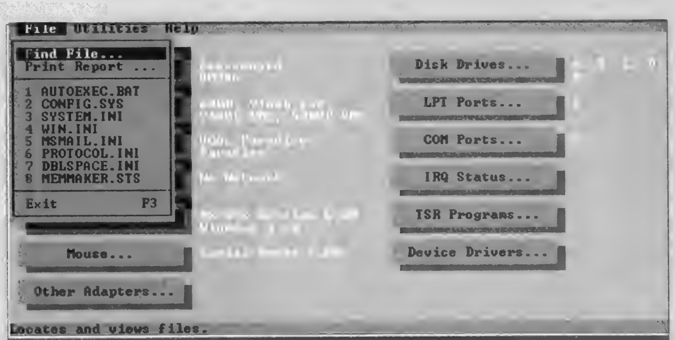
File menu. Lets you search for and display files, print a system-information report, view certain configuration files, and exit MSD. Lists several configuration files you can look at, including AUTOEXEC.BAT, CONFIG.SYS., WIN.INI, and SYSTEM.INI (Windows configuration files), DBLSPACE.INI (configuration for DOS 6's DoubleSpace disk compression), and MEMMAKER-.STS (the DOS 6 MemMaker status file). (See the accompanying screen below.) If you select a file that's not in your system, MSD returns you to the File menu.

Print Report. A File menu option. Lets you select only certain information or print the entire MSD display. You also can choose to include the contents of AUTOEXEC.BAT, CONFIG-.SYS, WIN.INI, SYSTEM.INI, and other configuration files.

Utilities menu. Includes two memory-viewing features. Memory Block Display shows the memory allocation of items in RAM, with their starting addresses and sizes in bytes; Memory Browser searches selected ROM and RAM areas for a text string you specify ("copyright" or "version," for instance). Also lets you print a test page to check your printer and switch from a color to a black-and-white display.

Insert Command. A Utilities menu option. Inserts a command in the appropriate file. The usefulness of this option is limited, however, because the only commands available are those stored in the MSD.INI file. Rather than edit MSD.INI to add a command and its file, it's easier to put the command into the relevant file yourself.

—K.J.



Locates and views files.

When you run MSD interactively, the File menu gives you easy access to the program's search feature. You may also view configuration files and generate system-information reports.

PC TOOLBOX

Put Some Life Into Your Batch Files

DOS treats your video display like a glass teletype. But use these Debug tools and your batch programs will take on a life of their own.

by Robert L. Hummel

For years, all my interaction with computers took place through a Model 55 Teletype (TTY) machine. I still have fond memories of the TTY's printer element bobbing as it traversed the roll of yellow paper, the rhythmic "chunka-chunka" of the hammer pounding out characters, and the smell of hot oil.

In contrast, personal computing today is a veritable Mardi Gras of visual stimuli. The pervasiveness of Windows, multimedia, and 17-inch color monitors with 1024-by-768-pixel displays makes it difficult to recall a time when computing wasn't so video-oriented.

But those of you who haven't enjoyed the thrill of working with a TTY can still appreciate the experience. Just imagine using an electric typewriter in place of your computer's video display. You can't run any applications that use graphics—remember, a typewriter is a character-based output device. And all your full-screen applications must go, since you can move only to the right and down.

Does this situation sound familiar? It should; you face it every time you use DOS. DOS treats your video display exactly like a glass teletype, writing text to the screen one character at a time, one line at a time. Type in a command line longer than 80 characters and DOS wraps automatically to the beginning of the next line of the display. As DOS writes the output from its commands, the screen scrolls up when full—just like a roll of paper.

Batch programs run under these same DOS constraints. Ignoring ANSI.SYS, batch programs have two options for cursor control: to write their output wherever the cursor happens to be, or to clear the screen and start at the beginning.

Is cursor manipulation possible in a batch program? As the old New England saying goes, "You can't get there from here." That is, not without a little help. Beginning in this issue, I'll present a set of video tools for your batch programs. These small utilities will improve your batch files' video output, making them look and perform better.

Nowhere to Go but Down

As I pointed out, DOS treats your video display as if it were simply a roll of paper. Your text scrolls up and out of sight as each new line is displayed on screen. And as text is written to the screen, the cursor is forced inexorably downward. There are times, however, when writing information blithely to the next line isn't the best choice for effective output.

Screen-based text editors, spreadsheets, and games are just a few examples of programs that benefit from a freewheeling cursor. This limitation on cursor movement is one of the biggest hurdles you must overcome if you're going to create snazzy batch programs. Fortunately, this limitation lies solely within DOS. The video-display routines built into your PC's BIOS (basic input/output system) make it easy to control your cursor.

I wrote the accompanying GOTOROW program to provide your batch

---

Contributing Editor Robert Hummel is the author of PC Magazine's Programmer's Technical Reference: The Processor and Coprocessor *and* PC Magazine Assembly Language Lab Notes *(both from Ziff-Davis Press).*

programs with a simple way to move up and down the screen. To convert the GOTOROW.SCR Debug script to an executable program, refer to the sidebar "Processing Debug Scripts."

Using GOTOROW is simple. When you want to move the cursor to a specific line on your display, use the following command:

```
GOTOROW row
```

where *row* is a valid line number. The first line on your display is always row 1. On the DOS default 80-by-25 text display, the last line will be row 25. If you've set your screen to use 43 or 50 lines with DOS's MODE command, however, GOTOROW will also accept row values up to those limits.

Try executing GOTOROW at your DOS prompt with different row numbers to move up and down the display. (This is a good way to while away those Friday afternoons waiting for 5PM.)

After a little experimentation, you may notice an interesting anomaly. Executing the command GOTOROW 1, for example, seems to leave the DOS prompt invariably on row 2. And executing GOTOROW 25 always scrolls the screen before displaying the prompt. The cursor winds up one row lower than you specify.

This isn't a bug in GOTOROW; the cursor moves to the correct row on screen. But DOS always moves to a new line before displaying its prompt. So even if you use GOTOROW to put your cursor on row 1, DOS will move to row 2, then display the prompt. In batch programs, where the DOS prompt doesn't normally appear between commands (when ECHO is off), this problem doesn't occur.

PROCESSING DEBUG SCRIPTS

To create any of these keyboard helper programs, you must first create the corresponding Debug script files shown. Using DOS's Edit function or an editor that saves files in ASCII format, type the instructions in the listing exactly as shown. Press the Enter key after each line, especially the *Q* in the last line. Also, be sure you enter a blank line just before the RCX instruction by pressing Enter twice. When you're done, save the file under the appropriate name and exit your editor.

Once you verify that you've created the file correctly, make sure Debug is in your current directory or in a directory listed in your PATH statement. Then create the executable program by typing the following command at the DOS prompt:

```
DEBUG < program.SCR
```

where *program*.SCR is the file you created with your editor. This command tells DOS to execute Debug and take its input from the SCR file instead of from the keyboard. When Debug encounters the Q instruction in the last line of the file, it terminates and returns control to DOS. If all goes well, you'll have created the program's executable COM file.

To make use of your new program, copy the COM file to the directory in which you store your batch or utility programs or into your DOS directory so that it will be available at the command line.

—R.H.

Use GOTOROW.SCR to create GOTOROW.COM, which provides your batch programs with a simple way to move up and down the screen.

```
N GOTOROW.COM
A 100
MOV     AH,F
INT     10
PUSH    BX
SUB     AX,AX
MOV     SI,5D
LODSB
CMP     AL,20
JZ      117
SUB     AL,30
AAD
MOV     AH,AL
JMP     10A
DEC     AH
MOV     DH,AH
SUB     DL,DL
MOV     AH,2
POP     BX
INT     10
RET

RCX
23
W
Q
```

End

DRG.BAT shows how you could use GOTOROW in a batch file.

```
@ECHO OFF
CLS
GOTOROW 10
ECHO  ****     ******    *****
GOTOROW 12
ECHO  **  ***  **   **  **     *
GOTOROW 14
ECHO  **   **  ******   **
GOTOROW 16
ECHO  **  ***  **  **   **   **
GOTOROW 18
ECHO  ****     **   **   *****
GOTOROW 11
ECHO  ******   *******  *******
GOTOROW 13
ECHO  **   **  **   **  **
GOTOROW 15
ECHO  **   **  *****    **  ***
GOTOROW 17
ECHO  ******   **   **  *******
GOTOROW 24
```

End

Use SAYAT.SCR to create SAYAT.COM. SAYAT.COM provides more flexibility than GOTOROW, letting you use special characters and position text more efficiently.

```
N SAYAT.COM
A 100
MOV     CL,80
DEC     CL
SUB     CH,CH
MOV     AL,22
MOV     DI,81
REPNZ
SCASB
JNZ     159
MOV     BP,DI
REPNZ
SCASB
JNZ     159
DEC     DI
MOV     AH,F
INT     10
PUSH    BX
SUB     AX,AX
MOV     SI,5D
LODSB
CMP     AL,20
JZ      12D
SUB     AL,30
AAD
MOV     AH,AL
JMP     120
DEC     AH
MOV     DH,AH
SUB     AX,AX
MOV     SI,6D
LODSB
CMP     AL,20
JZ      143
SUB     AL,30
AAD
MOV     AH,AL
JMP     136
DEC     AH
MOV     DL,AH
MOV     AH,2
POP     BX
INT     10
MOV     AH,40
MOV     BX,1
MOV     DX,BP
MOV     CX,DI
SUB     CX,DX
INT     21
RET

RCX
5A
W
Q
```

End

The program DRG.BAT shows a whimsical use for the GOTOROW utility. This program writes the letters *DRG* in large characters on your screen in interlaced fashion. Using GOTOROW, the cursor moves around the screen as first the even rows, then the odd rows, are written. (If you're using a fast PC, the interlacing action may take place too quickly for the effect to seem very dramatic.)

Does this program start some ideas perking in your head? More ambitious projects could include successive frames of a man walking across the screen—animation! Or a rotating marquée border around a menu. DRG.BAT shows everything you'll need to write your own impressive batch programs along these lines with GOTOROW.

A Disappearing Trick

GOTOROW doesn't check to ensure that you've specified a valid row number for your current display mode. This leads to some interesting applications.

Depending on your PC's BIOS, typing `GOTOROW 26` when you have only 25 lines on screen might make your cursor (and your DOS prompt) disappear below the bottom of your screen. While your cursor, and all DOS output, will be sitting happily on row 26, you won't be able to see any of it. (If your BIOS checks your screen setting, lines that would appear below line 25 scroll up to appear on the very last line of your screen.)

Regardless of the current row, DOS will still scroll the visible portion of your screen (rows 1 through 25) every time it starts a new line of output. Internally, DOS looks to see whether the current row plus one is greater than the number of rows on screen and, if so, scrolls the screen. DOS doesn't consider the possibility that you've moved the cursor below the visible area.

MENU.BAT shows how you can use SAYAT to position text on your screen.

```
@ECHO OFF
CLS
SAYAT  5 10 "KEY   FUNCTION"
SAYAT  7 10 "F1    HELP"
SAYAT  9 10 "F2   START PROGRAM"
SAYAT  2 35 "Main Menu"
SAYAT 23 30 "Enter Choice,
  Then Press Enter"
GOTOROW 24
```

End

CLIFHANG.BAT uses SAYAT to position your cursor in the middle of your screen.

```
@ECHO OFF
CLS
SAYAT 11 30 "Here's your prompt:"
SAYAT 13 30 " "
```

End

To restore your cursor to the visible portion of your screen, use the CLS command (forcing the cursor to row 1) or execute GOTOROW with a more reasonable row number.

You can use this disappearing trick to good advantage when you want to hide the output of a DOS command or a program such as CHKDSK. You could also build an elementary security program, moving the prompt off screen when soliciting a password. Or you can put up a menu and then hide the cursor while waiting for the user to enter a selection. (You can also play "hide the cursor" with your coworkers on slow days in the office, but make sure you choose your victims with care!)

Speech Control

The GOTOROW utility neatly eliminates DOS's obsession with moving your cursor down-screen. Combining it with the ECHO command lets you write text to any line you choose. Despite this, however, GOTOROW isn't the best tool for all cursor-movement jobs.

In some cases, you won't want to write an entire line of text. Putting a single character into column 80 with the ECHO command, for example, requires that you precede it with 79 blanks. Displaying this line has the effect of erasing everything in those first 79 columns—okay if that's your intent, but otherwise disastrous!

I wrote the accompanying SAYAT program to provide more flexibility than GOTOROW while eliminating the need for a separate ECHO statement. The syntax for SAYAT is:

```
SAYAT row column "string"
```

where *row* and *column* specify the screen coordinates where the text *"string"* is to be displayed. Valid values for *row* are the same as for the GOTOROW command. In most standard text modes, the leftmost column on your screen is column 1, and the rightmost is 80. The string to be displayed must be enclosed in quotes as shown.

Using SAYAT is more convenient than combining GOTOROW and ECHO. Not only can you use a single command, but you can write only the characters you require at the exact spot you want them. MENU.BAT shows an example of how you can use SAYAT to position text on screen. This program clears the display, then calls SAYAT to write five text strings to the screen. Finally, it calls GOTOROW to leave the DOS prompt on the last line of the display.

Of course, the MENU.BAT program won't really give you help or start a program when you press the indicated keys—it simply shows how to draw with SAYAT. To turn this demo into a fully functional menu program, you'd need to add a line to get a keypress from the user with the DOS CHOICE command or one of the keyboard tools featured in previous "Toolbox" columns. Your menu program could then launch different applications in response to the keys you press.

SAYAT has other benefits over ECHO. It can print characters (such as <, >, and |) that ECHO can't. And because it positions the cursor independently of DOS, you can use it to make the DOS prompt appear in otherwise unlikely places. The CLIFHANG.BAT program uses this technique to make the DOS prompt appear in the middle of your screen, rather than at the left edge.

BLANKER.BAT works with BLANKER2.BAT to create an animated screen blanker.

```
@ECHO OFF
REM Sample screen blanker
  using SAYAT
CLS
:LABEL1
FOR %%A IN (1 3 5 7 9 11 13 15
  17 19 21 23) DO CALL
BLANKER2
  %%A
GOTO LABEL1
```

End

Use this program, BLANKER2.BAT, with BLANKER.BAT.

```
@ECHO OFF
SET SPOT="-"
:LABEL1
FOR %%A IN (10 20 30 40 50 60
  70) DO SAYAT %1 %%A %SPOT%
IF %SPOT%=="/" GOTO LABEL2
IF %SPOT%=="|" SET SPOT="/"
IF %SPOT%=="\" SET SPOT="|"
IF %SPOT%=="-" SET SPOT="\"
GOTO LABEL1
:LABEL2
FOR %%A IN (10 20 30 40 50 60
  70) DO SAYAT %1 %%A " "
SET SPOT=
```

End

A Nifty Demo

Once you gain the power to move anywhere on screen and write any character, you can do nearly anything in a batch program—albeit in small, measured steps. Because I'm often accused of writing dull demos, and to show how easily you can cobble together some nifty programs, I hereby present BLANKER.BAT and its companion program, BLANKER2.BAT. Together they create an animated screen saver.

BLANKER uses a FOR loop to call BLANKER2 repeatedly with different arguments. In this case, the arguments represent row numbers. Each time it's called, BLANKER2 uses SAYAT to write a series of characters at seven different columns on the specified row. The five characters are written in the following order: –, \, |, /, and then a blank. The effect is one of twirling segments dropping down the screen. To stop the program, press Ctrl-Break.

Tapping Your Creativity

Batch programs are helpful, but are often unattractive. This issue, we've taken the first steps toward designing batch programs that both look and work better. With a little time and imagination, you can enhance your batch files with the two video tools presented here, GOTOROW and SAYAT. Alone or in combination, they'll help you create menus, draw screens, and add some elementary animation to your own programs. ■

Windows TIP

AUTOMATIC DOCUMENT DUPLICATION

When you use Windows' Write program to create a document, you may save a backup copy right along with the original document. Simply click on Write's File menu and choose the Save As option. When the Save As window pops up, select the Backup box. From then on, Write updates both the original and the backup document when you select the File menu's Save option.

Of course, if you close the document and reopen it later, you must choose the Save As option again and select the Backup box to keep your backup copy up to date.

—Lane Olinghouse

BATCH-FILE MEDIC

Turn Up the Volume

Believe it or not, you don't need assembly language or Debug scripts to write clever and powerful batch files. QBasic offers simple tools that can beef up DOS's batch-file language and let you check system information on the fly.

by Hardin Brothers

Many of the batch-file tricks described in *DOS Resource Guide* rely on assembly language and Debug scripts. Although they're useful, certainly, they may seem like magic unless you know the assembly language in which they're written.

But you don't really need to know assembly language to write powerful batch files. This month, I'll discuss two techniques that require only a little knowledge of DOS commands, the DOS batch-file language, and either QBasic or GW-Basic. Best of all, these batch files are easy to understand and adapt to your own needs.

Volume Labels

Ernie Bender of Vienna, Virginia, has a batch file that copies data to disks. He wants to use the disk's volume label to decide whether the correct disk is in the drive. If it isn't, he wants to be able to prompt the user and then check again until the user inserts the correct disk.

Technical Editor Hardin Brothers has been working with and writing about computers for 15 years.

This question is just one instance of a more complex problem. How can a batch file check system information such as disk labels, the amount of free space on a disk, the date or time, file sizes, or time stamps? All these pieces of information can be important in your batch files.

Batch files can read information from three sources: the command line, the environment, and errorlevels. Unfortunately, batch files can't read information directly from files, keyboard, memory, or any other source.

So the problem Ernie Bender and other readers face is how to translate information from one form to another. All the necessary information is available through DOS commands—for example, VOL displays the volume label—and you can find much of it from a Basic program. The trick is translating it from one source to another.

You can set the volume label when you format a disk or with the LABEL command. A volume label can be up to 11 characters long and can contain spaces and other special characters. DOS always translates labels to uppercase; many utility programs (including Symantec's Norton Utilities) can create volume labels in both upper- and lowercase characters.

If I type the following command:

```
VOL A:
```

on my computer right now, I see this message:

```
Volume in drive A is TEST DISK
Volume Serial Number is 0000-12FB
```

Actually, the message is preceded by a blank line, a fact we'll have to deal with later.

You can change the volume label with DOS' LABEL command. The serial number, which is derived from the system clock, won't change until the disk is reformatted. As we'll see, you can test either the label or the serial number to make sure the user has inserted the correct disk.

Our first and easiest task is to put the label and serial number into a file so that it's easier to manipulate. That takes a single command, issued either from the command line or from a batch file, like this:

```
VOL A: > LABEL.TXT
```

The right angle bracket sends the information to the file LABEL.TXT instead of to the screen. Once the label and serial number are in a file, we can choose one of two ways to process the information.

Weird Batch Files

Sometimes the best way to perform magic is to play "what if": What happens if you save the output of the VOL command in a batch file? That's simple; the first line of the batch file, which we'll call TESTVOL.BAT, looks like this:

```
VOL A: > TEMP.BAT
```

Now let's assume that the next line runs TEMP.BAT:

```
CALL TEMP.BAT
```

At first, that seems like a strange way to operate. How can you turn data into a useful batch file?

But let's watch COMMAND.COM as it tries to execute these two lines:

1. The first line of TESTVOL.BAT runs the internal VOL command and sends its output to a file called TEMP.BAT.
2. The CALL command in line 2 saves the current position in this batch file; you'll return to TESTVOL.BAT when TEMP.BAT is done.
3. Line 2 then starts TEMP.BAT.
4. COMMAND.COM then executes the first line of TEMP.BAT:

```
Volume in drive A is TEST DISK.
```

Wait a second. VOLUME IN DRIVE A IS TEST DISK doesn't look like a command at all. Well, at least not to us, because we immediately see it as data or as a statement. But COMMAND.COM doesn't see the line that way. As far as it's concerned, this line begins with the instruction VOLUME and ends with six parameters:

```
in
drive
A
is
TEST
DISK
```

To run the command VOLUME, COMMAND.COM searches the current directory and every directory in the DOS path for VOLUME.COM, VOLUME.EXE, and VOLUME.BAT, in that order. It will run the first program it finds.

Since no program or batch file called VOLUME exists, COMMAND.COM displays this error message:

```
Bad command or file name.
```

What if we create a program called VOLUME.BAT? TEMP.BAT will run it for us. And because we used the CALL command to start TEMP.BAT, control will return to the original batch file when VOLUME.BAT is done.

Here's one version of VOLUME.BAT that can handle spaces in the volume name:

```
@Echo Off
Shift
Shift
Shift
Shift
Set Disk_Label=%1 %2 %3 %4 %5 %6
```

The four SHIFT commands "throw away" the words IN DRIVE A IS. The last line stores the volume name in the environment variable called DISK_LABEL. The replaceable parameters %1 %2 %3 %4 %5 %6 will store the volume name even if it contains five spaces and looks something like this:

```
A B C D E F
```

The original batch file can then examine the variable with a command like this:

```
IF "%LABEL%"=="TEST DISK"
  GOTO Disk_Okay
```

See the first listing, Disk Label Reader, for an example of how you can roll all this into a single batch

DOS TIP LOOK BEFORE YOU DEFRAGMENT

Hard-drive optimization is a wonderful thing. DOS can access defragmented files more quickly, because it doesn't have to chase around after bits and pieces of files. The news about defragmenters isn't all good, however. After you run DOS's DEFRAG, or a similar utility, all the files that UNDELETE might have recovered are lost, unless you use the Delete Sentry option, which stores deleted files temporarily in a separate directory.

If you don't use Delete Sentry, you can save yourself aggravation by getting into the habit of looking through the names of deleted files *before* you defragment your hard disk. To do this, type the following line:

```
UNDELETE /LIST
```

Undelete all the files you might need and back them up to an archive disk. Then, and only then, is it safe to run DEFRAG.

—Herbert M. Dye, Jr.

Disk Label Reader. You can use a text file as a batch program that checks system information and displays it if you can translate that data into a readable source.

```
@Echo Off
REM  This batch file reads and displays
REM  the volume label of the disk
REM  in drive A.

REM  First write Volume.Bat, which will
REM  capture the volume label and place it
REM  in the environment.

Echo @Echo Off                       >  Volume.Bat
Echo For %%%%x In (1 2 3 4) Do Shift >> Volume.Bat
Echo Set Disk_Label=%%1 %%2 %%3 %%4  >> Volume.Bat

Echo.
Echo Place a disk with a volume label in drive A
Pause

REM  Now create LABEL.BAT
VOL A: > LABEL.BAT

REM  Run LABEL.BAT, which will run Volume.Bat
Call Label.Bat
Echo.
Echo. The volume label of the disk is %Disk_Label%
Echo.

REM  Finally, clean up.
DEL VOLUME.BAT
DEL LABEL.BAT
SET Disk_Label=
```

End

file. The program begins by creating VOLUME.BAT. Then it prompts the user to insert a disk into drive A and uses the VOL command to create LABEL.BAT. It calls LABEL.BAT, which in turn runs VOLUME.BAT to create the environment entry. After the program displays the disk label, it cleans up by deleting LABEL.BAT, VOLUME.BAT, and the DISK_LABEL environment variable.

There's one oddity in the program. It uses the ECHO command plus redirection to create VOLUME.BAT, which is a standard batch-file technique. If you want a percent sign to show up in the created batch file, however, you must include two percent signs in the original program.

The second line of VOLUME.BAT uses a loop to execute the SHIFT command four times:

```
For %%x in (1 2 3 4) DO Shift
```

Therefore, the original program must have four percent signs in a row to send two to VOLUME.BAT. Similarly, that first listing must use two percent signs in front of each parameter name in line 3.

You can use a similar technique to collect any information you can store in the first line of a data file, as long as the line will have a predictable first word. For example, it's often used to capture the current time or date.

This technique has two weaknesses. The first is that it can't detect consecutive spaces in the volume name. As far as it's concerned, TEST DISK and TEST   DISK are identical labels.

The second and more serious weakness is that this technique can't process the second line of the batch file. There's no way I know of to use it to collect the serial number instead of the disk label. You can change the disk label with the LABEL command, but you can't change the serial number without reformatting the disk.

If you use J.P. Software's 4DOS or its commercial cousin NDOS (included in The Norton Utilities) to replace COMMAND.COM, you can simplify the first listing considerably by using a variable function to find the disk label.

The second listing shown here, 4DOS/NDOS Label Reader, which runs only with either of those two programs, shows how a more powerful command processor can simplify batch-file programming.

Adding Power

No matter how you look at it, the batch-file language is a wimp. It's the 98-pound weakling of command languages, especially compared to such powerhouses as 4DOS, NDOS, and the REXX processor included in OS/2.

But if you don't have one of those programs, don't give up. GW-Basic (included with early versions of DOS) and QBasic (included with DOS 5.x and 6.x) can add enough power to the batch language that you may never need a better command processor.

I'll focus on QBasic here, but note that almost everything below also applies to the older GW-Basic and BasicA, if you're still using one of them. Just remember to add line numbers to your Basic program.

You can write a batch file, then call a QBasic program to do the work the batch file can't handle. A problem such as reading a disk serial number is almost trivial in QBasic.

What isn't so trivial is sending information from a batch file to QBasic and sending the results back from QBasic to the batch program. QBasic programs can't read command-line parameters, they can't set errorlevels when they're done, and they can't even store information in the environment for a batch file to read.

But it isn't really difficult to send information from a batch file to a QBasic program. The simplest way is to use the environment. In the batch file, you can create a new variable like this:

```
SET QB_INFO=Some info for QBasic
```

Then, in a QBasic program, you can use a command like this:

```
Info$ = Environ$ ("QB_INFO")
```

Once these lines have been executed, the QBasic string INFO$ will contain the value SOME INFO FOR QBASIC.

Another way to send information to a QBasic program is with a text file. In the batch file, you can place data in the file with commands like these:

```
Echo Some info for QBasic >
  QBASIC.INF
Echo More info for QBasic >>
  QBASIC.INF
```

As long as your QBasic program knows how to extract and use the information you send it, you can add as many lines as necessary to the QBASIC.INF text file.

Within the QBasic program, you'd open the file and read each line:

```
Open "QBASIC.INF" FOR INPUT AS #1
Line Input #1, Info(1)$
Line Input #2, Info(2)$
Close #1
```

Sending information back from a QBasic program to a batch file is much more difficult. You can try to set an environment string with a command like this:

```
Environ "QB_Reply=Info
  for batch file"
```

but that attempt is doomed to failure.

Any program, including a QBasic program, receives a copy, called a *child*, of the environment, not the original. If the program adds an entry, the new entry is also added to the copy of the environment. When the program ends, the copy is destroyed, and its memory is released. The entry QB_REPLY will disappear before the batch file can examine it.

A second problem with this programming approach is that DOS saves memory by limiting the size of the child environment. No matter how much space the environment contains, the copy sent to a program will occupy the minimum possible. DOS always stores the environment in a block of memory evenly divisible by 16. So the copy will contain zero to 15 bytes of free space when a program receives it.

Your QBasic program could return information in a one-line text file; you could then use the technique described in the first part of this article to extract that information. But this seems awkward and limits the QBasic utility to a single line of information.

A Better Solution

A smoother alternative is to make use of the odd way COMMAND.COM processes a batch file. Once it begins a batch file, COMMAND.COM opens the file, reads one line, closes the file, and then executes the line. It re-opens the file, reads the next line, closes the file, and then executes that line.

This is an inefficient process, to be sure, but it has one powerful advantage. As long as it doesn't change anything that has already been processed, a program can add more commands to a batch file that's running; COMMAND.COM will never know the difference.

For example, suppose you launch a QBasic program from a batch file that looks like this:

```
@Echo Off
QBasic /Run UTILITY.BAS
```

While the QBasic program is running, it can modify that batch file and add information. When it's done, the batch file might look like this:

```
Echo Off
QBasic /Run UTILITY.BAS
SET QB_Reply=Info for batch file
```

Using this technique, you can add as many lines of information as you want to the batch file. All that information will then be available when your batch file resumes.

It's difficult to add lines to the middle of a text file from a QBasic program, so one more step is necessary. The final example this month runs a QBasic program from a batch file called from the main batch file. The secondary batch file will set information in the environment for the primary batch file to use.

Putting It All Together

It's time to put all of this theory to work. The third listing presented

4DOS/NDOS Label Reader. A more powerful alternative: Using 4DOS or NDOS instead of COMMAND.COM simplifies the label-reading program.

```
@Echo Off
REM  This batch file reads and displays
REM  the volume label of the disk
REM  in drive A.
REM  You can use it only with 4DOS or NDOS

Echo.
Echo Place a disk with a volume label in drive A
Pause

Echo.
Echo. The volume label of the disk is %@Label[A:]
Echo.
```

End

Serial Number Reader. Combining QBasic utilities and batch files gives you two-way communication and lets you read a disk's serial number.

```
@Echo Off
REM  This batch file reads and displays
REM  the serial number of the disk
REM  in drive A.
REM  QBasic must be available in the current
REM  directory or in the DOS PATH for this
REM  batch file to work.

REM  First, write the QBasic program:
Echo Open "Label.Txt" for Input as #1     > DISK_ID.BAS
Echo Line Input #1, A$                    >> DISK_ID.BAS
Echo Line Input #1, A$                    >> DISK_ID.BAS
Echo Line Input #1, A$                    >> DISK_ID.BAS
Echo A$ = "SET Serial_Num=" + Mid$(A$,26) >> DISK_ID.BAS
Echo Open "DISK_ID.BAT" for Append as #2  >> DISK_ID.BAS
Echo Print #2, A$                         >> DISK_ID.BAS
Echo Close                                >> DISK_ID.BAS
Echo System                               >> DISK_ID.BAS

REM  Now write the batch file
Echo @Echo Off                            >  DISK_ID.BAT
Echo QBasic /Run Disk_ID.BAS              >> DISK_ID.BAT

Echo.
Echo Place a disk with a serial number in drive A
Pause

REM  Now create LABEL.TXT
VOL A: > LABEL.TXT

REM  Run DISK_ID.BAT, which will run DISK_ID.BAS
REM  Then display the results:
Call Disk_ID
Echo.
Echo. The serial number of the disk is %Serial_Num%
Echo.

REM  Finally, clean up.
DEL DISK_ID.BAS
DEL DISK_ID.BAT
DEL LABEL.TXT
SET Serial_Num=
```

End

here, Serial Number Reader, uses a pair of batch files and a QBasic utility to find the serial number of a disk in drive A.

Since this is just an example, the batch file displays the serial number on screen and then stops. In a complete program, you'd probably want to store the serial number from the first disk and then make sure the user hasn't changed disks while the batch file is running.

You could use a similar technique to test the amount of free space on a disk, the attributes of a file, or any other information your batch file requires.

The listing begins by creating the QBasic program and storing it in a file called DISK_ID.BAS. You could store the .BAS program separately, but your hard disk might soon become cluttered with many small files. I find it faster and easier to create short files as I need them and then erase them.

The QBasic program opens the LABEL.TXT file for input and then reads all three lines. The first line is blank, the second line contains the disk's label, and the third line contains the serial number, for which the program is searching.

The fourth line creates a batch-file command that will put the serial number into an environment variable. The next two lines store the command in the batch file that ran the QBasic program. Finally, the last two lines tidy up and end the program.

The second part of the listing creates the batch file called DISK_ID.BAT, which will run QBasic. This simple two-line batch file may seem unnecessary at first. But DISK_ID.BAT also provides a place for QBasic to write its SET commands, which will send information back to the main batch file.

The last parts of the third listing prompt you to insert a disk into drive A, copy the label and serial number to LABEL.TXT, and then run DISK_ID.BAT, which, in turn, runs DISK_ID.BAS.

Finally, the program displays the serial number and then erases the files and the environment variable it used.

Thanks to the techniques described here, you can write truly powerful batch files that don't depend on assembly language, Debug scripts, or any other sleight of hand. DOS programming doesn't have to be tricky; sometimes all you need are a few basic tools to get you started. ■

PRODUCT INFORMATION

4DOS 5.0, $69 plus $6 shipping
JP Software Inc.
P.O. Box 1470
East Arlington, MA 02174
800-368-8777
617-646-3975

The Norton Utilities 8.0, $179
Symantec Corp.
2500 Broadway
Santa Monica, CA 90404
800-441-7234
408-253-9600

BATCH-FILE MENUS

Pros' Choice: A Wealth of Options

The Dark Ages are over. Now you can create complex menu programs that run from a single, interactive batch file.

by Doug Lowe

Ever since DOS version 1.0, people have been using batch files to create simple lists, or menus, of programs that can be run by typing a number or letter. When you add the batch file's name to the end of AUTOEXEC.BAT, the menu appears whenever the computer is turned on, sparing the user the cold tyranny of DOS's command prompt.

Unfortunately, even the simplest of these batch files have been hampered by limitations in DOS's batch-file language. One of the most frustrating shortcomings was that earlier versions of DOS provided no way for the batch file to pause, wait for the user to type a key, resume execution, determine which key was pressed, and take an appropriate action (for example, run a specified program).

It's true that DOS has always had a PAUSE command, but PAUSE merely stops the batch file until the user touches a key—any key. Once that happens and the batch file resumes, there's no way to determine which key was pressed—a necessary function in any menu program.

It wasn't until DOS 6, which introduced the CHOICE command, that we gained a built-in way to make batch files respond to keystrokes. Before we get into the details of CHOICE, however, I want to review some of the ways people created batch-file menus before DOS 6.

Inadequate Workarounds

Over the years, DOS programmers have favored two basic approaches to creating batch-file menus. The first involves using several batch files whose names correspond to menu choices, such as 1.BAT, 2.BAT, 3.BAT, and so on. An accompanying menu batch file (usually named MENU.BAT) uses ECHO commands to display the menu choices, then ends, leaving the user at the DOS prompt. The user selects a menu choice by typing a number and pressing the Enter key. DOS then runs the corresponding batch file.

The MENU1.BAT program listing presented here (see page 40) shows a simple menu that uses this technique. Note that this listing and the others in this article probably won't run properly on your computer without changes in the application programs' names and paths. Batch files need to know where your applications are located before they can run them.

Note that each of the three smaller batch files ends with a command that reruns MENU1.BAT. That way, when the user exits a program, he or she returns to the menu.

The only drawback to this technique is that it exposes the user to the DOS command prompt. If he or she types an incorrect response, a "Bad Command or File Name" message appears and the display scrolls, sometimes moving part of the menu off screen.

The second common approach to batch-file menus is to use DEBUG to create a small program that pauses the batch file, waits for the user to press a key, and sets the DOS errorlevel code to indicate which key was pressed. The batch file can then use a series of IF commands to test the errorlevel and identify the key in question. *DOS Resource Guide* has

Contributing Editor Doug Lowe is the author of 15 computer books, including The Only DOS Book You'll Ever Need *(Murach),* The Least You Need to Know About DOS *(Murach), and* The Microsoft Press Guide to DoubleSpace *(Microsoft Press).*

published several of these key-retrieval programs, which usually have names like GETKEY or KEYGET. One of the more detailed explanations of how to create such programs appeared in a "PC Toolbox" installment called "Batch Files Without the Wait," by Robert L. Hummel (*DRG*#11, September 1993, page 29).

These techniques can get the job done, but they're awkward at best. Fortunately, the CHOICE command, introduced with DOS 6, makes it possible for batch files to process keystrokes quickly and cleanly. The result is top-quality batch-file menus.

Understandable Choice

Though CHOICE has been part of DOS for more than a year, many people stumble over exactly what it does and how it should be incorporated into a batch file. For an overview of CHOICE along with its switches and parameters, see the accompanying sidebar "Using the CHOICE Command."

USING THE CHOICE COMMAND

Like most DOS commands, CHOICE has a few switches and parameters you can use to customize its operation. Following is the complete syntax:

```
CHOICE [/C:keys] [/N] [/S] [/T:c,nn] [text]
```

The switches either change the way keystrokes are processed or change the prompt that appears on screen. Specifically:

/C:keys Tells CHOICE which keys the user may press to get a specific response. For example, /C:ABC lets you press A, B, or C. The colon is optional, so /CABC has the same effect. Without the /C switch, the default setting is /C:YN, giving you a yes/no choice.

/N By default, CHOICE lists all valid keys in the prompt you see on screen. Use /N to suppress this prompt.

/S Forces the CHOICE command to be case-sensitive. For example, the following command:

```
CHOICE /C:ab /S
```

will accept *a* or *b*, but not *A* or *B*.

/Tc,nn Lets you set a default keystroke choice (*c*) that will execute automatically after a certain number of seconds (*nn*) have passed. For example:

```
/TA,20
```

means to assume the user pressed A after 20 seconds.

text The prompt text displayed. If the text contains any unusual characters, or if you want to add an extra space to the end of the prompt, enclose the text in quotes:

```
CHOICE /N "Do you want to continue? "
```

—D.L.

MENU1.BAT is an example of the way menu batch files were written before the advent of DOS 6's CHOICE command. This method requires help from three additional batch files to load the programs selected from the menu.

```
@ECHO OFF
CLS
ECHO Main Menu
ECHO.
ECHO 1. Windows
ECHO 2. WordPerfect
ECHO 3. Lotus 1-2-3
ECHO.
ECHO Type a number,
  then press Enter.
```

End

1.BAT loads Windows, the program named in item 1 in the menu created by MENU1.BAT.

```
CD \WINDOWS
WIN
CD \UTIL
MENU1
```

End

2.BAT loads WordPerfect, the program named in item 2 in the menu created by MENU1.BAT.

```
CD \WP60
WP
CD \UTIL
MENU1
```

End

3.BAT loads Lotus 1-2-3, the program named in item 3 in the menu created by MENU1.BAT.

```
CD \123R34
123
CD \UTIL
MENU1
```

End

Like commercial and homemade keystroke-input utilities, CHOICE uses errorlevels—codes that a program sends to DOS to signal an error or to tell DOS that it has terminated—to transfer keystroke information to batch files.

The CHOICE command waits for the user to press a key, then sets an errorlevel to indicate which key was pressed. After the CHOICE command, the batch file should use IF commands to determine which key was pressed and to take appropriate action.

CHOICE lets the programmer limit the range of keys the user can press. For example, the following CHOICE command responds only to the numbers 1, 2, 3, and 4:

```
CHOICE /C:1234
```

Similarly, the next one responds only to the letters A, B, and C:

```
CHOICE /C:ABC
```

If the user presses any key other than those listed in the /C switch, CHOICE just beeps and waits for another keypress. As a result, batch files that use CHOICE don't have to loop to make sure the user presses a valid key, as the MENU2.BAT program listing (see page 42) does.

The errorlevel CHOICE issues doesn't correspond to the ASCII code of the key the user presses. Instead, it indicates which of the keys listed in the /C switch the user pressed. For example, for the following command:

```
CHOICE /C:ABC
```

the program sets the errorlevel to 1 if the user presses A, 2 for B, and 3 for C. For this command:

```
CHOICE /C:KMJ
```

the program sets the errorlevel to 1 if the user presses K, 2 for M, and 3 for J. In other words, the order in which you list the keys in the /C switch determines what each key's errorlevel setting will be.

The MENU2.BAT listing provides a simple menu constructed with the CHOICE command. A series of IF statements test the errorlevel CHOICE sets. (If this section confuses you, you'll find more information in the sidebar "Unraveling the IF Statement.")

The MENU3.BAT listing (see page 43) shows a more efficient way of doing the same job. It creates the menus shown in the accompanying screens (page 42). The batch file begins by displaying the Main Menu, then uses a CHOICE command to wait for the user's selection. The IF commands then direct the batch file to the appropriate label to handle the user's selection. The GOTO command at the end of the first section is there just in case CHOICE issues an errorlevel code of zero (0), which can happen if the user presses Ctrl-C.

Notice that the CHOICE command specifies the Esc key as a valid choice. If the user presses Esc while the Main Menu is displayed, it means that the user wants to exit to DOS. The Esc key must therefore be handled differently in the submenus, because there the preferred escape route is to the Main Menu, not directly to DOS.

If the user selects a program from the Main Menu (Microsoft Windows, WordPerfect, or Lotus 1-2-3), the instructions that begin after :LOTUS, :WORDPERFECT, and so on invoke the appropriate program. Notice how a GOTO START command follows the command that runs each program. Thus, when the user quits that program, MENU3.BAT resumes at the top and redisplays the Main Menu.

If the user selects option 4 from the Main Menu, the program displays the Utilities Menu, and the batch file waits again for the user's

UNRAVELING THE IF STATEMENT

Testing the errorlevel codes (sometimes called exit codes) emitted by DOS commands such as CHOICE is a source of many batch-file programming mistakes. For example, to check for an errorlevel of 1, common sense would suggest using an IF command such as this:

```
IF ERRORLEVEL 1 GOTO MENU1
```

While this command will send the batch file to the label MENU1 if the errorlevel code is 1, it will also send the batch file to MENU1 if the code is 2, 3, or any other value greater than 1.

This is because the IF ERRORLEVEL command actually performs a greater-than-or-equal-to test. Thus, in the above example, the batch file jumps to MENU1 if the errorlevel code is greater than or equal to 1.

When you use IF commands with CHOICE, you need to narrow things down more and test for single errorlevel settings. There are two ways to do that. First, you can test the codes in descending order, like this:

```
IF ERRORLEVEL 3 GOTO MENU3
IF ERRORLEVEL 2 GOTO MENU2
IF ERRORLEVEL 1 GOTO MENU1
```

With this method, because the higher-numbered choices come first, there's no way for the batch file to mistake a 3 for a 2.

Alternatively, you can pair IF ERRORLEVEL and IF NOT ERRORLEVEL commands to test for specific errorlevel settings:

```
IF ERRORLEVEL 1 IF NOT ERRORLEVEL 2 GOTO MENU1
IF ERRORLEVEL 2 IF NOT ERRORLEVEL 3 GOTO MENU2
IF ERRORLEVEL 3 IF NOT ERRORLEVEL 4 GOTO MENU3
```

I've seen both methods in action, and I know of no clear advantage one has over the other, other than that the first is shorter.

—D.L.

MENU2.BAT is a simple menu program that uses the DOS 6 CHOICE command to interpret keystrokes, thus avoiding MENU1.BAT's more cumbersome multiple-file method.

```
:START
@ECHO OFF
CLS
ECHO Main menu
ECHO.
ECHO 1. Windows
ECHO 2. WordPerfect
ECHO 3. Lotus 1-2-3
ECHO 4. Exit to DOS
ECHO.
CHOICE /C:1234 "Enter your
  choice: "
IF ERRORLEVEL 4 GOTO EXIT
IF ERRORLEVEL 3 GOTO 123
IF ERRORLEVEL 2 GOTO WP
IF ERRORLEVEL 1 GOTO WIN
:WIN
CD \WINDOWS
WIN
GOTO START
:WP
CD \WP60
WP
GOTO START
:123
CD \123R34
123
GOTO START
:EXIT
CLS
```

End

input. This time, the IF commands direct the batch file to the appropriate routines for handling the utility program the user selects. These routines are contained in the DAILYBU, WEEKLYBU, FORMAT, and SCAN sections. After each utility program is executed, a GOTO UTILS command sends the batch file back so that the Utilities Menu is redisplayed on top of the Main Menu.

Option 2 in the Utilities Menu leads the batch file to where the Format Menu is displayed. Once again, a CHOICE command waits for the user's input. Then the IF commands direct the batch file to the appropriate routine for handling the key the user presses.

If the user presses the Esc key here, the batch file jumps to the routine labeled :FORMATESC, the Main Menu is redisplayed, and execution jumps back to the :UTILS routine to display the Utilities Menu. Otherwise, the batch file is sent to the label A12 to format a 1.2MB disk in drive A, A360 to format a 360K disk in drive A, B144 to format a 1.44MB disk in drive B, or B720 to format a 720K in drive B. After each FORMAT command, a GOTO UTILS command sends the batch file back to redisplay the Utilities Menu.

I hope this gives you a good idea of how MENU3.BAT flows. In a later issue we'll take a second look at MENU3.BAT, examine some techniques I use to improve the appearance of the menus this batch file displays, and explore some additional options CHOICE offers DOS programmers. ■

Windows TIP

GROUP MENTALITY

If you use a group of related files on a daily basis, consider collecting them in a "work group." To do this, select New from the Program Manager's File menu and click on the New Program Group option. When the Program Group Properties dialog box appears, type an appropriate name into the Description box. Then open the File Manager and drag the icons (folders) of the related documents into the new work group. Make sure the documents have extensions that associate them with their parent programs. Microsoft Word and Ami Pro, for example, look for DOC files; Windows Write expects WRI files.

Thereafter, double-clicking on a document's icon, now located in the work group, runs the parent program and brings the chosen document to the screen, ready to edit.

—Lane Olinghouse

The simple on-screen menu created by MENU3.BAT (see page 43).

```
Main Menu

1. Windows
2. WordPerfect
3. Lotus 1-2-3
4. Utilities

Press Esc to return to DOS.

Enter your choice:
```

The Utilities Menu, which appears after the user chooses item 4 from the Main Menu.

```
Utilities Menu

1. Scan hard disk for
   problems
2. Format a floppy disk
3. Weekly backup
4. Daily backup

Press Esc to return
to Main Menu.

Enter your choice:
```

The Format Menu, which appears when the user choosees item 2 from the Utilities Menu.

```
Format Menu

1. Format 1.2MB in drive A
2. Format 360K in drive A

3. Format 1.44MB in drive B
4. Format 720K in drive B

Press Esc to return
to Utilities Menu.

Enter your choice:
```

MENU3.BAT uses the power of DOS 6's CHOICE command to process menu choices without requiring helper programs, as MENU1.BAT does.

```
@ECHO OFF
:START
CLS
ECHO.
ECHO Main Menu
ECHO.
ECHO 1. Windows
ECHO 2. WordPerfect
ECHO 3. Lotus 1-2-3
ECHO 4. Utilities
ECHO.
ECHO Press Esc to return to DOS.
ECHO.
CHOICE /C1234← /N Enter your choice:
IF ERRORLEVEL 5 GOTO EXIT
IF ERRORLEVEL 4 GOTO UTILS
IF ERRORLEVEL 3 GOTO LOTUS
IF ERRORLEVEL 2 GOTO WORDPERFECT
IF ERRORLEVEL 1 GOTO WINDOWS
GOTO EXIT

:UTILS
CLS
ECHO.
ECHO Utilities Menu
ECHO.
ECHO 1. Scan hard disk for problems
ECHO 2. Format a floppy disk
ECHO 3. Weekly backup
ECHO 4. Daily backup
ECHO.
ECHO Press Esc to return to Main Menu.
ECHO.
CHOICE /C1234← /N Enter your choice:
IF ERRORLEVEL 5 GOTO START
IF ERRORLEVEL 4 GOTO DAILYBU
IF ERRORLEVEL 3 GOTO WEEKLYBU
IF ERRORLEVEL 2 GOTO FORMAT
IF ERRORLVEEL 1 GOTO SCAN
GOTO EXIT

:LOTUS
CD \123R34
123
GOTO START

:WORDPERFECT
CD \WP60
WP
GOTO START

:WINDOWS
CD \WINDOWS
WIN
GOTO START

:DAILYBU
CD \DOS
MSBACKUP INCRC
GOTO UTILS

:WEEKLYBU
CD \DOS
MSBACKUP FULLC
GOTO UTILS

:FORMAT
CLS
ECHO.
ECHO Format Menu
ECHO.
ECHO 1. Format 1.2MB in drive A
ECHO 2. Format 360K in drive A
ECHO.
ECHO 3. Format 1.44MB in drive B
ECHO 4. Format 720K in drive B
ECHO.
ECHO Press Esc to return to Utilities Menu.
ECHO.
CHOICE /C1234← /N Enter your choice:
IF ERRORLEVEL 5 GOTO FORMATESC
IF ERRORLEVEL 4 GOTO B720
IF ERRORLEVEL 3 GOTO B144
IF ERRORLEVEL 2 GOTO A360
IF ERRORLEVEL 1 GOTO A12
GOTO EXIT

:SCAN
SCANDISK
GOTO UTILS

:FORMATESC
CLS
GOTO UTILS

:B720
CLS
FORMAT B: /F:720K
GOTO UTILS

:B144
CLS
FORMAT B:
GOTO UTILS

:A360
CLS
FORMAT A: /F:360K
GOTO UTILS

:A12
CLS
FORMAT A:
GOTO UTILS

:EXIT
CLS
```

End

RAM UPGRADES

Curing Short-Term Memory Maladies

Boosting your PC's RAM is the quickest, easiest way to handle memory deficiencies—short of buying a new machine. Before you operate, though, consider these five common questions.

by John M. Goodman

Just about anyone who can handle a screwdriver can add memory to a PC. Not only is plugging in RAM (random-access memory) one of the easiest, most powerful computer upgrades, but today's prices for memory chips and modules make it a bargain, to boot. Once you've installed the memory and configured your machine to take advantage of it, your PC will work harder and faster and do more for you than ever before.

Before you rush into things, however, a little practical advice is in order. First, not everyone with a slow-performing machine needs a RAM boost—your PC's setup and the software it uses may be partly responsible for the lag. In this piece, I'll provide some tips for identifying and solving these kinds of issues.

John Goodman has been teaching and writing about microcomputers for 15 years. His latest books are Memory Management for All of Us: Deluxe Edition *(Sams),* Hard Disk Secrets *(IDG Books), and* DOS 6.0 Power Tools, *with John Socha (Bantam Computer Books, now Random House Computer Books).*

Second, if you decide to upgrade your computer's memory, you need to know some specifics, such as how much RAM you already have, how to locate it, and what amount and type to buy. You may be wondering about where to buy it, too. I'll try to answer your pre-installation questions. (And, since you don't want to damage your system or yourself in the process, read carefully the precautions set out in the accompanying sidebar, "Handle with Care.") Armed with answers to the following common questions, you should be ready to flesh out your computer's memory.

How Do I Know When I Need More Memory?

Some computer users may be surprised to find that their "memory shortage" problem isn't a memory problem at all.

For example, you may actually have enough RAM, but your PC's setup isn't using it optimally. In that case, you may need only to adjust your start-up files (CONFIG.SYS and AUTOEXEC.BAT and, possibly, WIN.INI and SYSTEM.INI, if you're running Windows).

When you get a message warning "not enough memory," it often means the program you tried to run could not find enough memory in which to load itself and its data. Just because a program can't find the needed memory doesn't necessarily mean it isn't somewhere in your PC.

How does that happen? One possibility is that another program is using the memory you need. This means you have too many resident programs loaded. Programs listed in your CONFIG.SYS file, in lines beginning with DEVICE, DEVICEHIGH, INSTALL, or INSTALLHIGH, or ones listed in your AUTOEXEC.BAT file may be some of the culprits.

Only some resident programs grab memory and hold onto it; others take some memory while they're running and then relinquish it.

If your PC is running DOS version 5 or later, the DOS program MEM can show you which programs are resident in your PC. If you issue just the command MEM, you'll get a report on the total amount of memory in your machine (broken down into conventional and extended, plus possibly an amount of expanded memory) and information on how much of that memory is free (not currently in use). Adding a suitable command-line parameter after the word MEM and before you press Enter instructs the MEM utility to provide more details.

Run MEM /? to find out which command-line parameters (those letters you see after the forward-slash character) your DOS version accepts to display this information. For MS-DOS and PC DOS version 5 or later, the command form that gives you the easiest display of this information is MEM /C. It instructs DOS to show the memory used by each of the programs currently "in residence" within your PC.

Suppose you discover a lot of your PC's memory is in use before running a troublesome, memory-hungry program. One solution is simply not to load all that resident software. On the other hand, those programs are presumably loading because they do something useful for you. (Or at least the person who last modified your CONFIG.SYS and AUTOEXEC.BAT files thought they were going to do something useful.) Think about each one. Get help if you need to, but in some way find out precisely why each resident program is there.

If you need some of these resident programs only occasionally, and if you're using DOS 6, you can add a menu to your CONFIG.SYS file that lets you load each program only when you need it. If you're running an earlier version of DOS, various third-party utility programs can help. A good one is Boot Commander from V Communications. (See the accompanying "Product Information" box, page 50, for details.)

You probably do need more memory if you're running Windows or any other protected-mode program and it complains of too little memory. (*Protected-mode* programs, such as Windows, can access more than the first megabyte of the CPU's memory-address space; they can use up to 16MB of memory in a PC with an 80286 or 386SX CPU chip, while more advanced 386DX, 486, and Pentium CPU chips allow up to 4GB.)

Another good indication of RAM shortage is when you are running

HANDLE WITH CARE

When IBM shipped its original PC in 1981, the company included an installation manual with clear drawings and detailed instructions on how to open the case. IBM expected owners to add or change plug-in cards, memory chips, or a math coprocessor chip and perform other minor chores inside the system. Nearly every PC maker since has operated on the same assumption. (A few dealers put a seal on the case and honor their warranty only if you *don't* open the case. If that describes your PC, let your dealer add memory to it—at least until the warranty runs out.)

You have to be careful when you open your PC's case, however. When the case is closed, it provides protection for the delicate electronic parts inside. Once you go inside, you have to be sure you don't damage something, including yourself.

In addition to following the directions provided in your PC's manual, note these important precautions:

- Before doing anything, disconnect your PC's system unit from the wall socket or surge suppressor into which it's plugged. You must also disconnect any cables that go to another part of your system, which is itself plugged into the wall. (Or unplug that component from the wall, too.) Your monitor and printer are typical components that are plugged into the wall individually and are connected by a data cable to the system unit.
- Equalize your voltage and your PC's *each time* before you reach inside the case to touch something. When handling memory chips, pick them up while they are still in their static protective packaging. And, while you are holding that packaging, touch an unpainted part of the PC unit's frame or its power supply. Touch the case or power supply each time you start to reach inside. The idea is to make sure you don't zap any chips with static electric charges you may have acquired since the last time you discharged yourself. Don't worry that neither you nor the PC is at ground potential. Only differences in electric potential can cause a problem, and those go away as soon as you touch the PC's case.

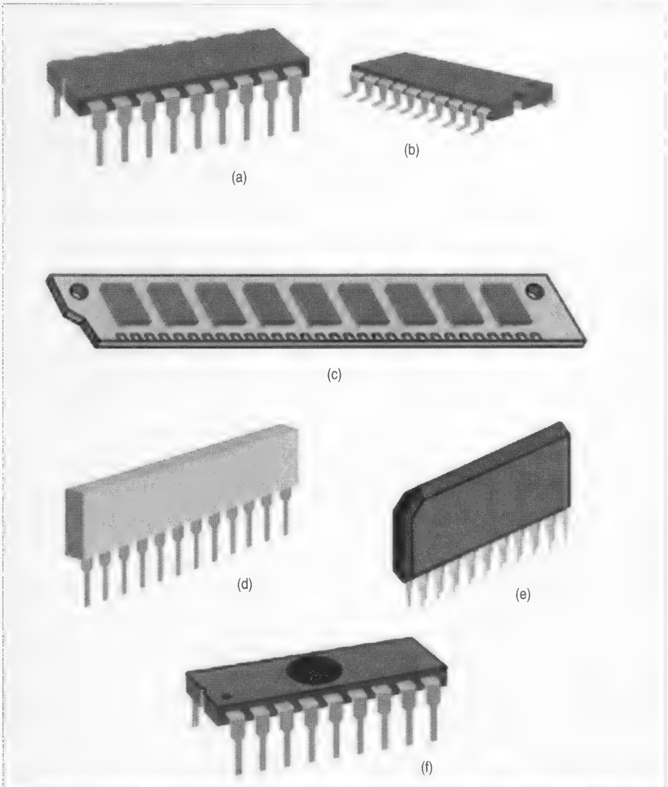
Here are a couple of other suggestions that might save you hours of headaches and much grief:

- Draw pictures of how cables are arranged inside your PC before you unplug any of them. Then use those pictures to be sure you plug them back in the same way.
- Double-check your work. Look very carefully with a bright light and a good magnifier to be sure that each pin on each memory module went into the socket hole you intended. It's easy to overlook the one pin that slips out around the socket or crumples up like an accordion, instead of going into the hole.
- Finally, before you replace the cover, hook up the power and other cables to the unit and test it briefly to be sure you put it back together correctly. If it works, shut it off, and then put the cover on.

—J.M.G.

Windows and you see your disk drive's access light flickering almost all the time. This usually means Windows is constantly swapping information out of memory to the disk to make more room in memory for other information. Since access to disk storage is about 10,000 times slower than access to electronic memory, such a process slows down progress considerably. Adding more memory is often the only way to stop Windows' constant swapping to disk.

These are just a few pointers on determining whether you need more memory. For more details, you can consult any of a number of good books on memory management. But, for the rest of this article, let's assume you have decided to add memory to your PC.


(a)
(b)
(c)
(d)
(e)
(f)

Several popular packages for memory chips include (a) DIP, (b) [illegible], (c) SIMM, (d) SIP, (e) ZIP, and (f) UV-EPROM

How Much RAM Do I Have Already?

You can get a good idea of how much memory you have simply by watching the screen while you start your machine. Either press the Reset button (if your PC has one), or else turn off the power switch, wait at least 20 seconds, then turn it on again.

At startup, most PCs perform a power-on self test (POST). One part of the POST tests the PC's memory. While it's running, a scrolling number on screen indicates how much memory it has tested and found good. (If you have an original IBM PC, it won't show you these numbers. Almost all other PCs and PC compatibles will, however.)

When it finishes, the final display will show either one number that represents the total amount of RAM in your PC, or it may be two numbers—one for so-called conventional memory and one for extended memory. (*Extended memory* is RAM above a PC's 1MB barrier; it carries its own unique memory addresses.) In the latter case, with extended memory in use, simply adding the two numbers gives you an idea of your PC's total memory.

Next, check this number against the one that should be on the sales receipt you got when you bought your PC; the two should be approximately the same. Sometimes, though, the number the POST reports doesn't include memory you know your PC has. There are at least two common reasons for this.

One is that many PCs hold up to 384K of memory for use as *shadow memory*. Here, part of your memory chips' capacity is reserved for possible use as stand-ins for some of your PC's read-only memory (ROM) chips. PC makers do this because a system can often read regular memory chips faster than certain ROM chips.

The other reason you might have bought more memory than the POST displays is that you've installed a hardware Expanded Memory Specification (EMS) card in your PC, which the POST doesn't test. Applications can't use this EMS memory until a special device-driver program activates it. Normally, that program is loaded by one of those CONFIG.SYS lines starting with DEVICE or DEVICEHIGH. When the device-driver program loads, it usually tests the expanded memory and reports how much it finds at the time.

What Type of Memory Do I Have?

Once you know how to determine the amount of memory in your PC, you need to find out what kind of memory chips or modules contain that memory. For that information, you can refer to the sales receipt, which might say (but often won't), or go to the horse's mouth, so to speak—open up your PC's case and look inside.

You probably received a manual with your PC that shows the inside of your system. Look at that manual carefully, and see where it shows your system's RAM is located. It also might note what kinds of memory chips or modules it uses and how much RAM you can install in your PC.

Most older PC models use small "orchards" of individual memory chips, each plugged into its own socket. One common arrangement is four rows with nine chips in each row. This arrangement is used with memory chips that hold one bit for each of a large number of addresses. The nine chips in a row hold nine bits (eight data bits and one bit, called *parity*, which checks the validity of the other eight bits) for each of that large number of memory addresses.

In another design, only three chips fit in a row. This happens when two of the three chips hold four bits for each of some number of addresses and the third chip holds just one bit for the same number of addresses. Again, the group of chips holds nine bits.

If your CPU chip is an 8088 (used in the original PC and XT) or an 8086 (used in the original PS/2, Model 30, and some clone XTs), your CPU refers to memory eight bits at a time. In that case, one of these rows is called a *bank*.

If your CPU chip is an 80286 (used in the AT) or a 386SX, it accesses memory two bytes (16 bits) at a time. In these computers, you need two rows of three or nine chips to make up a full bank of memory.

If the CPU is a 386DX, 486, or Pentium chip, or a chip that's compatible with one of those, it accesses memory four bytes (32 bits) at a time. In that case, you need four rows of chips to compose a full bank.

Usually, a PC motherboard has, at most, four rows of sockets for chips. Your PC also may have a plug-in card with more memory chips on it—an EMS card or additional extended memory. Again, this additional memory must total some number of complete banks.

NAVIGATING THE DOS MEMORY MAZE

by Robert Gryphon

Memory is a complex topic, and many PC users go through their computing lives with little appreciation for how it works. Most people pay attention to it only when a program complains that it's out of memory or when software simply won't load.

But knowing something about what computer memory is and how it operates is important—it can help you solve problems like this, and you'll also understand better how DOS utilities such as HIMEM.SYS and EMM386.EXE relieve memory shortages.

The 1-Megabyte Ceiling

When DOS was young, its designers set a 640K memory limit, which seemed like no limit at all. The PCs that dominated the marketplace at that time topped out at 48K to 64K, and no one was complaining. But as time wore on and programs grew larger, users demanded more, and program developers soon ran up against that 640K limit.

The original IBM PC uses an Intel microprocessor called the 8088. The limit to the amount of memory the 8088 can address isn't 640K; it's actually 1024K, or one megabyte (1,024 x 1,024 bytes = 1,048,576 bytes). The IBM PC and all its clones are designed to use certain regions of this memory range for specific purposes.

The range of memory from the very bottom (address zero) to 640K (address 655,359) is known as base memory or conventional memory; it's also called DOS memory, because it's the widely known 640K range that DOS uses for general purposes. It's the range of memory into which some or all of DOS itself is loaded, and it's also where programs are loaded when you start them.

You might ask why the memory range between 640K and 1024K wasn't also given to DOS for general use. Although DOS could have used it, in fact general-purpose RAM isn't the only type of memory a computer needs. The PC designates the memory between 640K and 1024K (known as upper memory) to support the operation of the computer.

For example, in addition to an address range for base memory (in this case, zero to 640K), your PC needs a place to store the BIOS—the basic input/output system, the programs that boot the computer and run it at the lowest level. It needs a place to store the current contents of the screen. (This region is often called video RAM.) And some adapter cards inserted into the system contain built-in memory that needs to be available to the processor.

Some of these regions, such as screen memory, are in the same location on pretty much all PCs. Other types of memory, such as the BIOS, vary in size and location, but are the same on all PCs of a given model. Still other memory, such as adapter RAM, can vary from system to system because the user configures it. All these types of memory compete for space in the upper memory area.

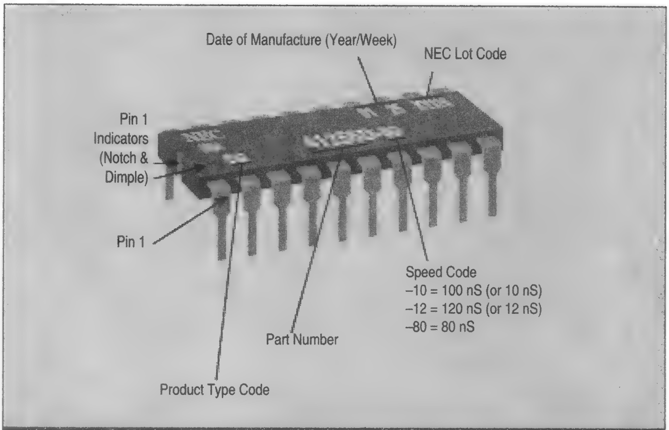
Onward and Upward to 16 Megs

IBM took the first step toward correcting the memory limit when it introduced its

Continued on page 49

Most PCs built in the last few years use memory *modules* instead of individual chips—small circuit cards with memory chips mounted on them. They plug into sockets. One common form is the *single in-line memory module*, or *SIMM*. (See the accompanying illustration, page 46.) You might have only one such module, or you might have as many as eight on the motherboard. More than that is rare.

Once you have found the memory chips, you need to determine what kinds they are. This information is contained in some tiny legends printed on the tops of the chips. You may need a strong light and a good magnifying glass to *see* those legends, let alone read them. Check all the chips—some may be different from the rest. Write down which types go into which sockets. Memory modules sometimes carry a part number on the circuit board. Other times you have to read the legends on the individual chips of each module to decide which type they are.



Here's how to read a typical memory chip's label.

The second illustration (above) shows a typical individual memory chip and how one manufacturer (NEC) indicated its type. Most other chip makers use a similar numbering and labeling scheme.

How Much Can I Install?

If your PC's motherboard has empty sockets for memory chips or modules, your easiest upgrade is simply to buy more identical chips or modules and install them in those sockets. If the sockets are filled, your PC might have as much memory as it can accommodate. Or it might not. Sometimes you can remove the existing chips or modules and replace them with new ones that have a larger capacity.

The only practical way to find out if this is possible is to check your PC's manual. If, for example, you have 2MB of RAM, and if that is in the form of two 1MB SIMMs, you might be able to add a couple more for a total of 4MB. Alternatively, you might be able to replace the original two 1MB SIMMs with 4MB SIMMs, and add two more of those for a total of 16MB.

Different kinds of CPU chips have different maximum amounts

DOS TIP

A TEMPORARY FIX

Check your AUTOEXEC.BAT file; if you see a line like one of the following, perhaps it's time you did a little editing:

```
SET TEMP=C:\DOS
SET TMP=C:\DOS
SET TEMP=C:\WINDOWS
SET TMP=C:\WINDOWS
```

By itself, using one of these lines isn't bad. But if you or someone else who uses your system habitually turns off the computer while an application or Windows is still running (or if you experience power fluctuations that make your system reboot), your hard disk will accumulate unnecessary temporary (TMP) files. If you don't check your hard disk periodically, you won't be aware of the problem.

To avoid wasting hard-disk space, use MD to create a directory for your temporary files. A logical choice is to store it one level below your C:\DOS or C:\WINDOWS directory (C:\DOS\TEMP or C:\WINDOWS\TEMP, respectively). Then edit your SET command. For example, if C:\DOS\TEMP is your directory for temporary files, change SET to read like this:

```
SET TEMP=C:\DOS\TEMP
```

Finally, insert an ECHO command such as this one immediately after your SET command:

```
ECHO Y|DEL < C:\DOS\TEMP
```

This line automatically deletes any files that reside in C:\DOS\TEMP at bootup.

Note that the ECHO command is language specific. If DOS is loaded in a language other than English, you must use the appropriate replacement for the Y parameter. In French, for instance, you should use O (*oui*).

—Larry Peace

of memory they can address. The original PC uses an Intel 8088 chip that can address only a little more than a million bytes of information. (We call that amount 1MB.) Later models with 80286 or 386SX chips can address up to 16MB. Still more advanced models use the 386DX, 486, or Pentium chips, which can address up to 4096MB (also called 4GB).

Just because your PC has a chip that can address thousands of megabytes, though, you still can't install all the memory you want. Aside from the difficulty of affording them all, you must have some place to plug in the memory chips or modules.

Typically, modern PCs come with sockets for anywhere from 4MB up to around 64MB of memory. The only way to go beyond your PC motherboard's limit is to plug in a card with more memory sockets on it. Often, this is not a good idea, as memory on such a card might not be as speedy as main motherboard memory. The only common exception is if you use a special, proprietary add-on memory card sold by your PC's manufacturer.

You *must* add memory in complete banks, however. If you have rows of nine sockets for individual memory chips, each row must be either full or empty. Furthermore, if your PC uses an 80286 or 386SX CPU, you must have an even number of rows filled. And if your CPU is a 386DX, 486, or Pentium, you have to fill four rows of chips at a time. Your PC will work best if all the chips in any one bank are the same brand as well as the same type.

If your PC uses the industry standard form of SIMM, they hold 1MB each. (Actually they hold 9 bits of information at each of 1,048,576 locations. The industry jargon for these is "1Mx9" SIMMs.)

One of these SIMMs could be a complete bank, but only for PCs with 8-bit memory access. The second group of CPUs, whichaccess memory 16 bits at a time, must have pairs of SIMMs. The most advanced CPUs, with their 32-bit memory access, require four of those SIMMs per bank.

Some PCs, especially ones made by IBM, use nonstandard SIMMs that hold 2MB or 8MB per module. These SIMMs are 18 bits wide (including the parity bits), so you need only half as many SIMMs per bank.

Don't throw away any memory chips you remove; they're still valu-

NAVIGATING THE DOS MEMORY MAZE

Continued from page 47

PC/AT, which contains an 80286 processor. The 286 is designed to address up to 16 megabytes of memory. This additional RAM is called extended memory, and it opens the door to a whole new order of large and complex applications.

But to remain compatible with the tremendous number of existing 8088- and the similar 8086-based systems, DOS itself wasn't redesigned to work with the new memory limits. In other words, DOS programs still assumed that the BIOS and video RAM were in their original locations in upper memory.

That doesn't seem like a severe problem at first. You might say, just use the base memory to run applications, leave the upper memory alone, and give all the new extended memory to DOS. But it wasn't that simple. DOS was tied to the 8088 chip, and all DOS-based programs were designed around the 8088's method of addressing memory. Existing DOS programs had no idea how to use the memory range above that first megabyte.

A few special-purpose utilities included with DOS could make some use of the extra RAM. These were the original disk-cache and RAM-disk programs, which included new built-in memory management techniques. Such programs are initially loaded into base memory; then they map large ranges of extended memory and access it in simple ways.

But computer users still wanted better (and larger) programs. To provide an alternative solution, Microsoft synthesized the industry's needs and ideas into its Extended Memory Specification (XMS), a driver that would make extended memory easier for programmers to access. While the existing class of DOS-based programs would still address only the original 640K, XMS opened the door to a new class of DOS-based programs that could use as much extended memory as they wanted.

Microsoft and IBM included an XMS driver with DOS 4.x: HIMEM.SYS. This driver lets programs access extended memory and also another special region, the high-memory area (HMA), which technically is the first 64K of extended memory. It's an interesting but little-understood fact that the 80286 processor (and higher processors) can actually address 1088K even when acting like an 8086 processor. Thus, in addition to the first megabyte, the 64K of memory above the first megabyte is directly addressable by DOS programs. Only one program can use this region of memory at a time. At first, only third-party utility programs exploited the HMA, but now with Microsoft's HIMEM.SYS, DOS and any number of other programs can use it.

Although XMS solved many problems for users of more advanced personal computers, a lot of 8088/8086-based PCs were still in use. Because these computers couldn't address extended memory, users were stuck at the 640K limit. The demand for a solution resulted in the formation of a consortium of vendors, who came up with a brand-new memory-addressing scheme that would let older computers use more memory. This method, the Expanded Memory Specification (EMS), required a special memory board and a DOS device driver.

Like the XMS driver discussed above, an EMS driver provides a set of functions for programmers that makes the new memory relatively easy to address in their code. Older DOS programs not designed for EMS wouldn't access it, but EMS became popular enough that most important programs began to use it.

Continued on page 50

able. Keep them until you know your PC's new chips are working perfectly. Then you either can sell the old ones or put them in another machine. Some laser printers, for example, are able to print more complex documents or print them faster if you add memory to them.

Is All RAM Alike?

Memory chips and modules come in a confusing variety of models. To increase the memory in your PC, you usually want some kind of *dynamic random-access memory*, or *DRAM*. But even this RAM type breaks down into others, such as *static column DRAM*, *fast page-mode DRAM*, and so on. But you don't need to learn all those terms. What you need are the exact brand and model recommendations from your owner's manual or the legends on your PC's existing chips or modules. Then consult a memory-chip vendor. Explain what you have and get pricing for exact replacements.

The vendor also might offer cheaper chips produced by another manufacturer. They can be a bargain if the seller guarantees the chips will work in your PC (and if that guarantee includes letting you return them if they do not). Watch out for nonequivalent SIMMs. A careless vendor, for example, might just try to sell you 1Mx8 SIMMs instead of 1Mx9. They won't work unless yours is one of the relatively few PCs that doesn't support a parity bit per byte of main memory.

To a degree, memory chips or modules are like aspirin. They're commodity products, and you may buy them from the lowest-price source you can find. Remember, though, that unlike aspirin, memory may have to be returned. So make sure the retailer or the mail-order vendor stands behind the product and will exchange it or refund your money if you're not satisfied.

Once you match the memory chips you need with the price you're willing to pay, you'll be ready to treat your computer—and yourself—to a real power boost. ■

NAVIGATING THE DOS MEMORY MAZE

Continued from page 49

Four Gigabytes—and Climbing?

When 80386 computers were introduced, programs managing the 386 in a certain way could address up to 4 gigabytes of RAM: 4,096 megabytes, or 4,294,967,296 bytes. Hardware manufacturers had finally found an unreachable limit—at least for the near future.

At that time, the trend in memory chips was shifting toward SIMMs (single in-line memory modules) and away from individual memory chips. SIMMs are little circuit boards that contain 256K, 1 megabyte, 4 megabytes, or more of RAM each; you insert them as modules into an appropriate PC.

Memory addresses by nature must be unique, so these overlaps must be resolved. The natural inhabitants of upper memory supersede extra RAM in their respective ranges. There's an interesting aftereffect, though. In between the regions used by components such as video RAM and BIOS are regions of unused RAM. These areas are known as upper-memory blocks, and, of course, fall within that first megabyte of memory DOS can use.

At first, DOS ignored these free memory blocks. Some programs, such as Norton's System Information, would point them out as free memory, but they weren't contiguous with the first 640K, so DOS's standard program loader couldn't use them. Utility programs such as Quarterdeck's QEMM changed all of that. QEMM was the first of a number of programs that, by replacing DOS drivers, was able to utilize those upper memory addresses.

Microsoft liked QEMM and similar products so much that it designed these functions into its next edition of DOS, version 5. First, Microsoft added a utility called EMM386.EXE, a device driver that runs with HIMEM.SYS. EMM386.EXE duplicates some of QEMM-386's upper-memory block mapping and expanded memory management. In addition, Microsoft added the new CONFIG.SYS command DEVICEHIGH, which loads device drivers into upper-memory blocks, and the internal DOS command LOADHIGH, which loads TSR (terminate-and-stay-resident) programs into upper-memory blocks.

Incidentally, DOS 5 also addressed users' complaints that DOS 4.x was a memory hog. Microsoft redesigned the main DOS kernel (the part that loads when you boot up) to load partially into the high-memory area—that first 64K of extended memory. You can configure your installation of DOS 5 or higher to do this by specifying the line `DOS-HIGH, UMB` in your CONFIG.SYS file. (This works only with a 286 or higher processor.)

Because of Microsoft's efforts as well as those of a number of other software designers, DOS users no longer suffer from the same memory constraints that hampered performance in the past. A number of clever techniques and hardware solutions have helped DOS maintain its usefulness throughout many generations of software.

Even so, it's important to understand that DOS's memory troubles have always stemmed from its addiction to the memory-addressing method of the old 8086 and 8088 microprocessors. The final remedy, of course, will be a redesigned DOS. ■

Robert Gryphon is a free-lance writer and programmer/analyst in Redmond, Washington. He welcomes comments and questions; you can reach him on CompuServe at 70541, 3420.

PRODUCT INFORMATION

Boot Commander 1.2, $59.95
V Communications Inc.
4320 Stevens Creek Blvd.
Suite 275
San Jose, CA 95129
800-648-8266
408-296-4224

SYSTEM COMMUNICATION

Understanding Your PC's BIOS

It's the all-important mediator between hardware and software, and knowing how it works can unlock your computer's deepest secrets.

by John Bryan

Your PC is defined, to a very large degree, by its basic input/output system: The BIOS determines how the hardware interacts with the world. And while you can get along for years without knowing how your computer's BIOS works, familiarity with its basic functions can help you troubleshoot system failures, set up a new peripheral, or begin exploring hardware-level programming.

Fundamental BIOS

A computer initializes its components when you first turn it on; that is, the master controller sends out an electrical signal and waits for an appropriate response, much as you might flip a switch and expect a light to come on. Once the computer is initialized, it's ready to operate.

You might visualize a PC as a series of functional layers. (See the accompanying illustration on the following page.) At the uppermost layer are the applications, the programs you use to do your daily work. Beneath these programs is the system software, which includes the machine's operating system and the device drivers. The system software provides basic functions for applications, letting them interact with the hardware.

Hardware is the lowest system level—the ground floor, so to speak—where all the actual work is done. You don't need applications or an operating system to access hardware, but they make your computing life much easier. The system BIOS provides a sort of insulation layer between the hardware and the software.

None of the software layers is perfectly stratified, of course; interaction occurs at all levels. The BIOS is software, too, but it's written into the hardware, and so is called "firmware." That is, the BIOS exists even when your computer is off. The essence of the BIOS's job is to provide primitive input/output services and hardware interrupt handling.

A PC is controlled largely through the use of interrupts. That is, the hardware will keep on doing whatever you or your software tells it to do until something stops it. An interrupt can come from an application, from system software, or from peripheral hardware such as a printer. When the processor receives an interrupt, it relinquishes control of the system to an interrupt handling routine. Your PC's BIOS controls most of these routines, regardless of the interrupt's source.

Each interrupt carries a unique number, INT *xx*. The number *xx* is in hexadecimal (base 16). Besides the decimal digits zero through 9, hex adds A through F, representing 10 through 15. Base-10 15 is hex F (FH), base-10 16 is 10H, base-10 17 is 11H, and so on. In working with systems where a word is 16 or 32 bits, hexadecimal is much easier than base 10.

Interrupt numbers range in value from 00H to FFH, which provides for 256 possible assignment numbers. IBM originally established the conventions regarding the way interrupt numbers are assigned, parceling them out to the processor, other hardware, system software, and application software,

---

John Bryan is a freelance writer in San Jose, California. His background is in electrical engineering and computer marketing.

and keeping some of them reserved for unspecified future purposes.

By establishing this "map" of how the PC works, IBM spurred the development of the entire compatibles industry. Now, many manufacturers besides IBM make system BIOSes, and several companies, including Phoenix Technologies and American Megatrends, have made a business of supplying BIOSes to third-party vendors who may not have the technical resources or interest in making their own.

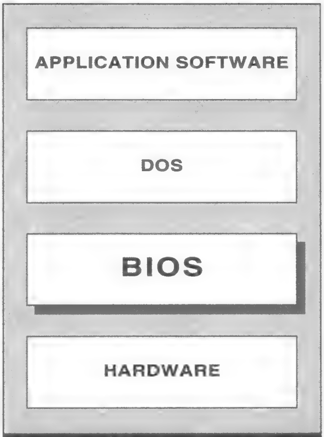
The POST Position

But what exactly does the BIOS do? When you turn the computer on, the BIOS starts its power-on self-test (POST). Based on instructions written into its code, the BIOS sends out initialization signals to all the hardware components in the system, making sure that each one responds appropriately and performing a basic function test.

One of the components the BIOS examines is memory, and you see a message on screen telling you that the test is underway. Then the BIOS initiates the operating-system bootstrap, which loads DOS, by invoking the INT 19H bootstrap loader. If you start the POST by turning on the machine or pressing the reset button, the routine is exactly the same, but if you reset by pressing Ctrl-Alt-Del (INT 09H), the POST doesn't check memory above 64K.

Interestingly, the POST still checks for the cassette storage system that was an option on the original PC and XT. Some peripheral vendors now use this hardware interrupt as a *hook*—an entry point to the system—where they can attach their products, such as special-purpose disk controllers, scanners, or other equipment. More common peripherals usually have an established position in the interrupt schema.

While the BIOS performs its POST, it also creates the interrupt-vector table in system memory. Vectors direct an interrupt signal to the appropriate hardware or software that will handle it. Once the BIOS creates the table in RAM, it *initializes*, or makes ready for use, the vector addresses of its own hardware interrupt-handling routines. Once your system loads DOS, the operating system initializes those vectors that are reserved as its own. Other hardware elements perform the same task, initializing reserved interrupts; application software, when you load it, will follow suit.



The computer's basic input/ouput system (BIOS) is a program, stored on a chip on the motherboard, that provides DOS and application software with essential information about the underlying hardware.

Toward the end of the POST process, the system BIOS looks for optional read-only memory. The function of optional ROM is to extend the BIOS's operations beyond its original capacity. Optional ROMs exist for every VGA card on the market and are often found in conjunction with disk peripherals such as ESDI or SCSI hard disks, and CD-ROM. The system BIOS looks only in specified areas for this extra ROM, and the expansion ROM, if any, must be configured in a specific manner.

Once the POST finishes testing and initializing and all system ROM is ready to go, the BIOS then invokes INT 19H. This starts the part of the POST that searches drive A and then C for a bootable disk. If the POST finds one, it reads the boot sector (a section of the disk containing a small program for starting DOS) and then writes its contents to system memory. If it doesn't find a boot sector, the POST falls back on INT 18H, which will generally invoke an error-message routine.

In the bad old days of personal computing, "No boot device available" was a relatively common message, but systems, and especially disks, are much more stable now. INT 18H may also be used to send the BIOS looking for a remote boot, such as that used in a network's diskless workstation.

The Stash

The BIOS also stores, via its setup program, all pertinent information concerning your computer. (See the accompanying sidebar, "Elephant's Memory: A Tour of BIOS Setup Routines.") This data is stashed in CMOS memory via the INT 15H device service routine. The BIOS gets some of this information, such as the amount of system memory installed, via its POST; other data, such as the time or perhaps the disk type, you have to enter yourself.

Once you've input this data, the BIOS uses the system battery to keep it active even when the system itself is shut off. If your system battery dies, however, an on-screen message from the BIOS tells you that you'll need to run this setup routine again.

When a system is booted properly and running, it's the BIOS's job to arbitrate and handle interrupt service routines and device service routines. The combination of these tasks represents just about everything a PC does, so you can see that the BIOS is integral to the smooth functioning of every operation in your system. (For a look at how QBasic and assembly-language programs use interrupts, see "More Programming Power to You," *DRG*#15, May 1994, page 58.)

What's Firmware?

The standard BIOS is burned into an erasable, programmable read-only memory chip. A hardware technician can erase standard EPROMs with ultraviolet light, admitted through a special window into the chip. That's also why a label always covers the window when the chips are in use: Stray ultraviolet light, while it won't erase the chip, can change it enough to fail. While there's little random ultraviolet light bouncing around inside the average computer, of course, in these days of a deteriorating ozone layer it's best not to take chances.

Once a ROM has been erased, a program can be installed by applying voltage to the external leads, or legs, of the chip with an EPROM burner. This device reads data from a software program and translates it into the proper sequence of electrical pulses to the chip leads.

The voltage applied to the chip during this process is much higher than the level used by the system where the chip will be installed; common voltage levels are 12.5V and 21V. (Most system logic is based on a 5V standard).

You'll also see references to electronically erasable PROMs. An EEPROM can be erased by electricity, though it's generally still done at higher voltage than standard system logic. The difference between EPROM and EEPROM is largely a matter of the chip's construction materials. Some EEPROMs can be programmed at 5V (American Micro Devices, among others, makes them), but most manufacturers use a higher programming voltage.

Flash EPROMs, an Intel invention, are EEPROMs made of yet a different material. The essential difference between an EEPROM and Flash is largely a matter of the number of process cycles each can withstand. When a chip is erased and reprogrammed repeatedly, it begins to suffer mechanical fatigue, a phenomenon not normally associated with electronic components, but one that does occur nonetheless. In fact, developing a material that could withstand an appropriate number of write cycles has been one of the major stumbling blocks to developing solid-state disks—storage devices that would have much faster access times than hard disks.

Upgradable BIOSes

Besides solid-state storage, Flash EPROMs and EEPROMs are also the types of ROM chips used for BIOSes you can upgrade, as found in many of today's systems. Upgrading has several benefits over the traditional method of pulling the ROM BIOS chip and replacing it.

First, you don't need to worry about destroying a new chip while installing it. The legs of ROMs are easily bent or broken. Also, as with all chips, you have to insert ROMs a specific way, or they either won't work or will burn up.

Second, you don't have to reestablish all the ROM BIOS information already installed in the computer's setup program. With an upgradable ROM BIOS, that information is never altered, so you don't have to replace it.

DOS TIP

CUT IT OUT!

Here's a time-saver that makes quick work of removing a single line from any text file. It's a trick you may use right at the DOS prompt; no word processor or text editor is required.

Start by issuing the TYPE command to view the text file you want to modify. This command, for instance, displays CONFIG.SYS on screen:

```
TYPE CONFIG.SYS
```

(Whenever you plan to alter CONFIG.SYS, it's a good idea to make a backup copy first.) DOS then displays the contents of the file, which might look like this:

```
BUFFERS=10
FILES=40
DEVICE=SETVER.EXE
LASTDRIVE=Z
DOS=HIGH
```

If the line you want to delete is DEVICE=SETVER.EXE, type the following at the DOS prompt:

```
TYPE CONFIG.SYS | FIND /V "DEVICE=SETVER.EXE" > CONFIG.SYS
```

Presto! The unwanted line is plucked from the file.

This works because, instead of displaying CONFIG.SYS, DOS feeds the output of the TYPE command to FIND, which acts as a filter. The /V (for *variant*) parameter tells FIND to display only lines that don't contain the string DEVICE=SETVER.EXE. The output redirection symbol (>) sends the resulting text, cleansed of the DEVICE line, back to CONFIG.SYS and overwrites the original file.

You need to be careful, however. When choosing the contents of the FIND string, you must match the use of upper- and lowercase letters exactly. If you don't, DOS won't remove the line in question from the file. Remember, too, that this technique removes all lines containing the string you specify.

—John Wolfskill

A side benefit is that you experience virtually no downtime on your computer. If you're like most computer users today, you use your system to get a job done, and taking the case apart and playing with the chips on the motherboard isn't high on your list of priorities. With an upgradable BIOS, the process takes about two minutes if you have the software on disk, or maybe 15 minutes if you get it from a company bulletin-board system.

But chances are you've never needed to change the BIOS in your system, and that may always be so. For the first eight or nine years of the history of the PC, changes came slowly, at least the ones that might have required a change in the BIOS.

Much of the concern over upgradable BIOSes comes from the portable market, where changes have come quickly. Just a few short years ago, a typical portable with an 80286 CPU weighed 15 to 20 pounds, had a CGA-quality monochrome liquid-crystal display, and didn't include a hard disk. Today, a portable with an 80486 CPU may weigh less than five pounds, sport an active-matrix color VGA screen, and come with more than 100MB of disk capacity.

The drive to bring 486 power and high-quality displays to a lightweight machine meant that battery life had to be conserved as much as possible. This, in turn, led to the development of intelligent power-saving peripherals and power-management software that would turn off portions of the system not in use. The problem was that the operating system and the BIOS weren't always able to tell whether an initialization signal came from an active device that required a

ELEPHANT'S MEMORY: A TOUR OF BIOS SETUP ROUTINES

The BIOS searches the computer, via the power-on self test, in its initial check of system components. If it recognizes any changes from the information it keeps stored in CMOS memory, or if this is the first time you've turned on the computer, the BIOS will warn you to run the system-setup routine.

System-setup routines vary among manufacturers, but they have many features in common. The most popular way to access the setup routine is to hit the Del key during the BIOS POST. Many systems also offer a hot-key option, such as Ctrl-Alt-S, to switch to the setup routine whenever the system is at a DOS prompt.

If the BIOS finds some fundamental change in the system during its POST—for instance, a difference in amount of system RAM—it will prompt you to go to setup and adjust the values stored in CMOS. Some newer systems now offer a plug-and-play BIOS; if the POST notes a change, it notifies you, but makes the changes itself. SystemSoft uses this method in its BIOS, and others will probably use the same technique.

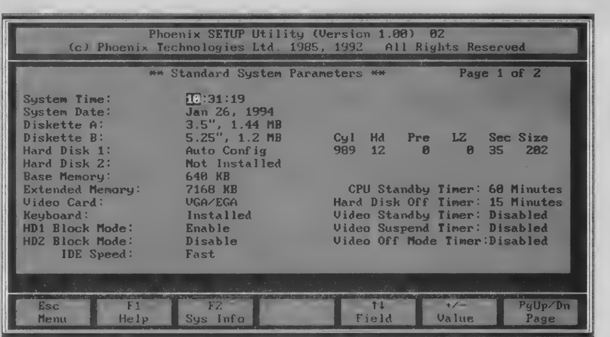
Once in the setup routine, you'll usually see a menu of system options you can change. The menu may include, but isn't limited to, standard and advanced CMOS options, disk utilities, chip-set options, automatic configuration, password utilities, and display utilities (found in integrated VGA and notebook systems, usually). You also have the option to save the information or exit without saving.

Choosing any of these options will most often produce a warning, if you haven't seen one yet, that any changes you make may affect the function of the computer seriously.

Note that making setup changes isn't for the uninformed. Always check your manual or consult your system's technical-support staff unless you're sure of what you're doing. If you're uncertain, exit the setup routine without saving. If you do decide to change the information, first write down the system CMOS settings, especially the hard-disk type.

The standard CMOS setup screen will show all the basic settings for your system, including type of processor and system clock speed, time and date (it may have an option for incorporating daylight savings), number and type of floppy drives, number and type of hard disks, amount of system and video memory, whether the system has a math coprocessor, and perhaps whether the system has a keyboard installed and what language format that keyboard supports.

Hard-disk options are usually keyed to an extensive table of standard types (47, or perhaps more) with a couple of user-configurable types. Most IDE drives and high-capacity ESDI (pronounced *ez-dee*) and SCSI (*scuzzy*) drives will be in this



```
                Phoenix SETUP Utility (Version 1.00)  02
     (c) Phoenix Technologies Ltd. 1985, 1992   All Rights Reserved

            ** Standard System Parameters **           Page 1 of 2

System Time:            10:31:19
System Date:            Jan 26, 1994
Diskette A:             3.5", 1.44 MB
Diskette B:             5.25", 1.2 MB       Cyl  Hd   Pre   LZ   Sec Size
Hard Disk 1:            Auto Config         989  12     0    0   35   202
Hard Disk 2:            Not Installed
Base Memory:            640 KB
Extended Memory:        7168 KB               CPU Standby Timer: 60 Minutes
Video Card:             VGA/EGA             Hard Disk Off Timer: 15 Minutes
Keyboard:               Installed           Video Standby Timer: Disabled
HD1 Block Mode:         Enable              Video Suspend Timer: Disabled
HD2 Block Mode:         Disable             Video Off Mode Timer:Disabled
     IDE Speed:         Fast
```

The CMOS setup program lets you reprogram the BIOS to alter your PC's memory, video display, and other hardware. You can usually access it by pressing a special key combination, such as Ctrl-Alt-Esc. Shown here is the main setup screen for a Gateway 2000 4DX-33V: a 33MHz 486-based computer equipped with a Phoenix Technologies BIOS.

response or from one that was kept inactive to conserve battery power.

Finally, the recent explosion of multimedia applications and hardware has also introduced changes, in both portable and desktop computers. CD-ROMs, sound cards, accelerated video, 24-bit color video, and other changes are making their presence felt. The possibility that the system BIOS might need updating is more real than it might have been even a year or two ago.

Equipping your system with an upgradable BIOS isn't a modification you can make at home. And unless your system was designed to support it, you can't do it at all. To reprogram your BIOS in-system, you need special circuitry on the motherboard. Part of it is a DC-to-DC converter that will take the system logic's voltage from 5V to 12V. This circuit is tied into the BIOS circuitry, so when programming is needed, it can be done in place. In addition, the existing BIOS must be specially designed to contain hooks to save the state of all of its registers and memory in a separate place while reprogramming is going on.

More and more often, manufacturers are offering this feature on new models, especially notebooks. This trend is a boon for buyer *and* vendor. The buyer knows his or her machine can handle upgraded peripherals and software, and the vendor gains a loyal customer.

Chances are, you'll never have to interact directly with your computer's BIOS ROM. Still, it's comforting to know enough about how it works that you'll be able to cope more effectively with the occasional backup-battery failure or a peripheral upgrade that doesn't work. ■

user-configurable "slot." This is because the drive configuration information is on the drive itself, and the information in CMOS is basically a pointer to the drive.

Advanced CMOS setup, if your system has it, may include many options, along with elaborations of the basic options. You may be able to change the system-clock rate at bootup, or toggle between high and low speeds.

There may also be support for non-Intel math coprocessors such as the Weitek. You can usually skip the memory test or turn off the Num Lock key when you turn on the computer. If your system has a cache, you'll probably be able to disable it (generally for testing purposes), and you may be able to select your boot device (skipping drive A, for instance).

One of the more significant features of advanced CMOS is the option to cache BIOS ROM in RAM, moving often-used, hard-coded instructions from relatively slow memory to very fast memory. This can speed up the performance of video, disk, and any other components that use ROM, as well as the system itself. This copying of ROM to RAM is called *shadowing* and can usually be done in 8K to 64K blocks, depending on the size of the ROM you want to cache.

Besides the CMOS setup, most systems will also feature a setup for chip-set options. This element is a range of choices you can access if you want to manipulate the system chip set, which controls many input/output (I/O) functions and memory operations in your system.

Options here may include selecting I/O bus speed as a function of the system clock; the number is generally derived by dividing the clock speed by 2, 3, or 4. Selecting a faster speed, such as "divide by 2" in a 33MHz system, which yields a 16.66MHz bus, will result in significantly faster performance from bus-mounted peripherals (This is the concept behind local bus).

Note that some I/O cards won't work at these speeds, though, so it may take some experimenting to determine the optimal level for your system.

Memory options, also controlled by the chip set, might include settings for memory-refresh rates, wait states for memory and cache read and write cycles, cache buffer output, and fast decode.

Settings such as those mentioned above will customize your memory area by matching the system clock with whatever speed RAM you have installed. If your RAM is faster, wait states may be lower or zero, while slower, less expensive RAM may require two or more wait states for correct operation.

You should change disk-operation settings rarely, as most of them are destructive to the contents of the disk. These include formatting, setting the disk interleave, and media analysis of the disk surfaces.

In most cases, you're probably better off going with a third-party disk product, such as Symantec's Norton Utilities (version 8.0, $179; Symantec Corp., 2500 Broadway, Santa Monica, CA 90404, 800-441-7234 or 408-253-9600) or something similar, which will give you more flexible options, non-destructive file optimization, and additional file- and data-recovery features.

Video operations are much the same. For the most part, unless you have a portable computer, you won't use this option beyond the initial setting. With a portable system, the changing environment may mean that you'll have more use for changing your video settings.

An exception to the general rule—which is "Set 'em and leave 'em alone"—is when the system backup battery dies, either of old age (anywhere from two to ten years) or for reasons unknown. In either case, this is one time when you'll want to have all your original settings in writing. If you do, you can just reenter the values and off you go. If you don't, you may lose the contents of your hard disk, as well as be forced to spend a significant amount of time fooling around with the system to optimize it again.

—J.B.

QBASIC ENTERTAINMENT

Mad Bomber

An evil scientist has hidden 15 explosives within a 10-by-10 grid. Can you find them all fast—without blowing yourself to bits? A little luck, a lot of skill—it's classic arcade fun.

by Doug Lowe

Remember that scene in *Raiders of the Lost Ark* where Indiana Jones works his way cautiously across the room to retrieve a priceless artifact—while carefully avoiding the booby-trapped floor tiles? One wrong step and he's history.

My QBasic program, Mad Bomber (see the accompanying listing), works on a similar premise. Hidden within a 10-by-10-square playing field are 15 bombs. Your job is to pinpoint the location of each one without "stepping" on any of them. If you keep your head, you'll come out a winner, but make one mistake and the game's up.

Want to try your hand? You have two options. If you have a modem, download the listing from the *DRG* bulletin board (BBS). (See the accompanying sidebar, "Downloading BOMBER.BAS," for instructions.) If you're modemless at present but do have MS-DOS 5.0 or later, you can fire up QBasic and type in the listing. It takes longer than downloading, but you get the chance to learn a little about programming in QBasic as you go.

Creating the Program

To create BOMBER.BAS, start QBasic by typing `QBASIC` at the DOS prompt. Type a few lines of the program, then use the File menu's Save option, and type `BOMBER.BAS` when it asks for a filename. As you type in the rest of the program, save your work frequently. When you finish typing in the main program, save the file one last time.

BOMBER.BAS includes three subprograms—INITIALIZEGAME, SHOWGRIDITEM, and SHOWNEIGHBORS—which you may create in either of two ways.

The first method is to type in the main program, then type `SUB` to mark the beginning of each subprogram. Then press Enter, and QBasic whisks you to a separate window, where you type each subprogram. QBasic supplies an END SUB command automatically to mark the end of each subprogram; you type the lines between SUB and END SUB. The second way to create a subprogram is to use the Edit menu's New Sub option. (The main program must be loaded for this procedure to work.) After you supply a subprogram name, QBasic sends you to a new window in which the beginning SUB and ending END SUB lines appear. As in the other method, you type the lines between these two commands.

Whichever method you use to create the subprograms, be sure the opening SUB line for SHOWGRIDITEM reads as follows:

```
SUB SHOWGRIDITEM (X!, Y!, MODE)
```

and that the opening line of SHOWNEIGHBORS reads:

```
SUB SHOWNEIGHBORS (X, Y)
```

The program won't function properly unless these lines include the parameters provided within parentheses.

As you type in the program, you may need to move from subprogram to subprogram or between a subprogram and the main program. To do

---

Contributing Editor Doug Lowe is the author of 15 computer books, including The Only DOS Book You'll Ever Need *(Murach),* The Least You Need to Know About DOS *(Murach), and* The Microsoft Press Guide to DoubleSpace *(Microsoft Press).*

DOWNLOADING BOMBER.BAS

To connect to the *DOS Resource Guide* BBS, set your modem to 8 data bits, no parity, 1 stop bit (8, N, 1), then dial 1-603-924-3181. (Drop the 1-603 if you're calling from within New Hampshire.) Opening messages will ask you to supply a first name, last name, and password. If you're a new user, the system also asks you to fill out a questionnaire.

With the preliminary procedures completed, you're free to browse around the system, leave messages, download, or upload. To download BOMBER.BAS, choose option F from the main menu and then D from the Files menu. At the "File #1?" prompt, type `BOMBER.BAS`, press Enter to tell the system you're ready to download, and then follow your communication program's standard procedures for downloading a file. If you answered *Y* to the "Automatically log off after last download is completed?" prompt, the connection will be broken as soon as the download is finished. If not, return to the main menu and select option G to log off.

—Eds.

that, press F2. Select the program module you want from the list QBasic provides, and press Enter.

One more note on subprograms is in order. When you save a program that includes a subprogram, QBasic adds the necessary DECLARE statement automatically to the beginning of the program. Don't be suprised when the final program contains DECLARE statements you didn't type in.

Bombs Away!

To load QBasic and run BOMBER.BAS, type `QBASIC /RUN BOMBER` at the DOS prompt. If you prefer to start the game by typing `BOMBER`, create a batch file named BOMBER.BAT that invokes QBasic and BOMBER.BAS, and put it into a directory listed in your PATH statement. You may also start the program by running QBasic, choosing the File menu's Open command, typing `BOMBER.BAS`, and selecting the Start command from the Run menu, or pressing Shift-F5.

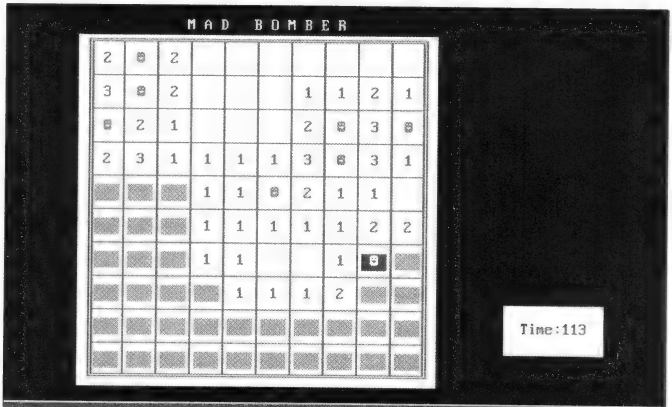
BOMBER.BAS starts by clearing the screen and drawing the playing field. To play the game, you uncover squares on the grid one by one, using the arrow keys to move to a particular square and highlight it. If you don't think the highlighted square contains a bomb, press the Enter key to reveal its contents; press the spacebar to mark a square as the site of a bomb.

Play continues until you press F10 to quit the game, press F2 to begin a new round of Mad Bomber, pinpoint correctly the locations of all 15 bombs, or press Enter on a square within which a bomb is hidden.

As you uncover squares that don't contain bombs, the game rewards you with clues. A blank square indicates that the eight squares surrounding the empty one (three above, three below, and one on either side) are free of bombs. A numeral (1 through 8) tells you the number of bombs housed in the eight surrounding squares.

For example, if you uncover a square containing a 2, you know that two of the surrounding eight squares hold bombs and six don't. (See the accompanying screen shot of a Mad Bomber game in progress.) The trick, of course, is figuring out *which* of the surrounding squares contain the bombs. To win you'll need to rely on old-fashioned deductive reasoning, with an occasional calculated risk and a bit of luck.

The results of your first few moves depend on chance. In fact, because 15 of the 100 squares contain bombs, there's a roughly 1:15 chance that you'll lose the game on your first move. If your luck holds and you uncover two or three squares successfully, you should accumulate enough clues to begin pinpointing bombs and uncovering empty squares.



Mad Bomber in progress. Sometimes a lucky move will uncover half the grid; then again, you have a 1 in 15 chance of losing with your first step.

BOMBER.BAS, a.k.a. Mad Bomber. To win, you must uncover 15 bombs hidden within a 10-by-10 grid.

```
DECLARE SUB SHOWNEIGHBORS (X, Y)
DECLARE SUB SHOWGRIDITEM (X!, Y!, MODE)
DECLARE SUB PLAYGAME ()
DECLARE SUB INITIALIZEGAME ()
CLS : SCREEN 0
DIM SHARED GRID(11, 11)    '0-8 = adjacent bombs;
  9=bomb
DIM SHARED SHOW(11, 11)    '0 = hidden
  1 = exposed  2 = marked
DIM SHARED OPENX           'Number of squares
  currently revealed
DIM SHARED FIRSTMOVE, STARTTIME
CONST NUMBOMBS = 15        'Number of bombs

RANDOMIZE TIMER
CALL INITIALIZEGAME
ENDGAME = 0
OPENX = 0
MARKS = 0
U$ = CHR$(186)
S$ = STRING$(17, 32)
S5$ = STRING$(5, 32)
S4$ = STRING$(4, 32)
S3$ = STRING$(3, 32)
X = 1: Y = 1: CALL SHOWGRIDITEM(X, Y, 1)
DO
    COLOR 8, 3
    DO: IN$ = INKEY$
        IF FIRSTMOVE = 0 THEN
            LOCATE 19, 68: PRINT USING "###";
              TIMER - STARTTIME;
        END IF
    LOOP WHILE IN$ = ""
    IF IN$ = CHR$(13) THEN      'Pressing Enter
      reveals square
        IF FIRSTMOVE = 1 THEN
            STARTTIME = TIMER
            FIRSTMOVE = 0
        END IF
        SHOW(X, Y) = 1
        CALL SHOWGRIDITEM(X, Y, 1)
        IF GRID(X, Y) = 9 THEN
            COLOR 8, 3
            FOR X = 1 TO 10: FOR Y = 1 TO 10
                SHOW(X, Y) = 1
                CALL SHOWGRIDITEM(X, Y, 0)
            NEXT Y: NEXT X

            COLOR 15, 4: BEEP

            LOCATE 5, 58: PRINT CHR$(201) +
              STRING$(17, 205) + CHR$(187);
            LOCATE 6, 58: PRINT U$ + S$ + U$;
            LOCATE 7, 58: PRINT U$ + S5$ +
              "Bummer!" + S5$ + U$;
            LOCATE 8, 58: PRINT U$ + S$ + U$;
            LOCATE 9, 58: PRINT U$ + S4$ +
              "You Lost." + S4$ + U$;
            LOCATE 10, 58: PRINT U$ + S$ + U$;
            LOCATE 11, 58: PRINT U$ + S3$ +
              "Press a key" + S3$ + U$;
            LOCATE 12, 58: PRINT U$ + S4$ +
              "to begin" + S5$ + U$;
            LOCATE 13, 58: PRINT U$ + S3$ +
              "a new game." + S3$ + U$;
            LOCATE 14, 58: PRINT U$ + S$ + U$;
            LOCATE 15, 58: PRINT CHR$(200) +
              STRING$(17, 205) + CHR$(188);

            DO: LOOP UNTIL INKEY$ <> ""
            CALL INITIALIZEGAME
            X = 1: Y = 1: CALL SHOWGRIDITEM(X,
              Y, 1)
            OPENX = 0: MARKS = 0
        ELSE
            OPENX = OPENX + 1
            IF GRID(X, Y) = 0 THEN
                CALL SHOWNEIGHBORS(X, Y)
                CALL SHOWGRIDITEM(X, Y, 1)
            END IF
            IF OPENX = 100 - NUMBOMBS THEN
                COLOR 8, 3
                FOR X = 1 TO 10: FOR Y = 1 TO 10
                    SHOW(X, Y) = 1
                    CALL SHOWGRIDITEM(X, Y, 0)
                NEXT Y: NEXT X
                COLOR 15, 1: BEEP
                LOCATE 5, 58: PRINT CHR$(201) +
                  STRING$(17, 205) + CHR$(187);
                LOCATE 6, 58: PRINT U$ + S$ + U$;
                LOCATE 7, 58: PRINT U$ + S3$ +
                  "Excellent!" + S4$ + U$;
                LOCATE 8, 58: PRINT U$ + S$ + U$;
                LOCATE 9, 58: PRINT U$ + S4$ +
                  "You Won." + S5$ + U$;
                LOCATE 10, 58: PRINT U$ + S$ +
                  U$;
                LOCATE 11, 58: PRINT U$ + S3$ +
                  "Press a key" + S3$ + U$;
                LOCATE 12, 58: PRINT U$ + S4$ +
                  "to begin" + S5$ + U$;
                LOCATE 13, 58: PRINT U$ + S3$ +
                  "a new game." + S3$ + U$;
                LOCATE 14, 58: PRINT U$ + S$ +
                  U$;
                LOCATE 15, 58: PRINT CHR$(200) +
                  STRING$(17, 205) + CHR$(188);
                DO: LOOP UNTIL INKEY$ <> ""
                CALL INITIALIZEGAME
                X = 1: Y = 1: CALL SHOWGRIDITEM(X,
                  Y, 1)
                OPENX = 0: MARKS = 0
            END IF
        END IF
    ELSEIF IN$ = " " THEN
      'Spacebar toggles mark
        IF SHOW(X, Y) = 0 THEN
            SHOW(X, Y) = 2: MARKS = MARKS + 1
        ELSEIF SHOW(X, Y) = 2 THEN
            SHOW(X, Y) = 0: MARKS = MARKS - 1
        END IF
        COLOR 8, 3: LOCATE 25, 77: PRINT USING
          "###"; NUMBOMBS - MARKS;
        CALL SHOWGRIDITEM(X, Y, 1)
    ELSEIF LEN(IN$) = 2 THEN
        SELECT CASE RIGHT$(IN$, 1)
            CASE CHR$(77)             'Right arrow
                CALL SHOWGRIDITEM(X, Y, 0)
                IF X < 10 THEN X = X + 1
                CALL SHOWGRIDITEM(X, Y, 1)
            CASE CHR$(75)             'Left arrow
                CALL SHOWGRIDITEM(X, Y, 0)
                IF X > 1 THEN X = X - 1
                CALL SHOWGRIDITEM(X, Y, 1)
            CASE CHR$(72)              'Up arrow
                CALL SHOWGRIDITEM(X, Y, 0)
                IF Y > 1 THEN Y = Y - 1
                CALL SHOWGRIDITEM(X, Y, 1)
            CASE CHR$(80)              'Down arrow
                CALL SHOWGRIDITEM(X, Y, 0)
                IF Y < 10 THEN Y = Y + 1
                CALL SHOWGRIDITEM(X, Y, 1)
            CASE CHR$(60)              'F2
                CALL INITIALIZEGAME
                X = 1: Y = 1: CALL SHOWGRIDITEM(X,
                  Y, 1)
                OPENX = 0
            CASE CHR$(68)              'F10
                COLOR 7, 0: CLS
                SYSTEM
            CASE ELSE
        END SELECT
    END IF
LOOP UNTIL ENDGAME = 1
'This marks the end of the main program. Type in
'the following three subprograms as described in
'the accompanying article.

SUB INITIALIZEGAME

LC$ = CHR$(32) + CHR$(186)
RC$ = CHR$(186) + CHR$(32)
BLC$ = CHR$(32) + CHR$(200)
BRC$ = CHR$(188) + CHR$(32)
LCC$ = CHR$(32) + CHR$(199)
RCC$ = CHR$(182) + CHR$(32)

UG$ = STRING$(3, 205) + CHR$(209)
MG$ = STRING$(3, 32) + CHR$(179)
MLG$ = STRING$(3, 196) + CHR$(197)
BG$ = STRING$(3, 205) + CHR$(207)
SL$ = CHR$(32) + CHR$(201)
SR$ = CHR$(187) + CHR$(32)
ST$ = STRING$(3, 205)
BK$ = STRING$(3, 32)

UGRID$ = "": MGRID$ = ""
MLGRID$ = "": BGRID$ = ""

FOR J = 1 TO 9
    UGRID$ = UGRID$ + UG$
    MGRID$ = MGRID$ + MG$
    MLGRID$ = MLGRID$ + MLG$
    BGRID$ = BGRID$ + BG$
NEXT J

COLOR 7, 0: CLS
COLOR 10: LOCATE 1, 23: PRINT "M A D   B O M B E R"
COLOR 8, 3
LOCATE 2, 10: PRINT SL$ + UGRID$ + ST$ + SR$;

FOR X = 3 TO 19 STEP 2
   LOCATE X, 10: PRINT LC$ + MGRID$ + BK$ + RC$;
   LOCATE X + 1, 10: PRINT LCC$ + MLGRID$ +
     STRING$(3, 196) + RCC$;
NEXT X

LOCATE 21, 10: PRINT LC$ + MGRID$ + BK$ + RC$;
LOCATE 22, 10: PRINT BLC$ + BGRID$ + ST$ + BRC$;
```

```
FOR X = 0 TO 11: FOR Y = 0 TO 11: GRID(X, Y) = 0:
  NEXT Y: NEXT X
FOR BOMB = 1 TO NUMBOMBS
    BOMBPLACED = 0
    DO: X = INT(RND * 10 + 1): Y = INT(RND * 10 + 1)
        IF GRID(X, Y) = 0 AND GRID(X, Y) <> 11 THEN
            GRID(X, Y) = 9
            BOMBPLACED = 1
        END IF
    LOOP UNTIL BOMBPLACED = 1
NEXT BOMB
FOR X = 1 TO 10: FOR Y = 1 TO 10
    IF GRID(X, Y) = 0 THEN
        COUNT = 0
        IF GRID(X - 1, Y - 1) = 9 THEN COUNT =
          COUNT + 1
        IF GRID(X, Y - 1) = 9 THEN COUNT = COUNT + 1
        IF GRID(X + 1, Y - 1) = 9 THEN COUNT =
          COUNT + 1
        IF GRID(X - 1, Y) = 9 THEN COUNT = COUNT + 1
        IF GRID(X + 1, Y) = 9 THEN COUNT = COUNT + 1
        IF GRID(X - 1, Y + 1) = 9 THEN COUNT =
          COUNT + 1
        IF GRID(X, Y + 1) = 9 THEN COUNT = COUNT + 1
        IF GRID(X + 1, Y + 1) = 9 THEN COUNT =
          COUNT + 1
        GRID(X, Y) = COUNT
    END IF
    SHOW(X, Y) = 0
NEXT Y: NEXT X

'Show grid
FOR X = 1 TO 10: FOR Y = 1 TO 10
    CALL SHOWGRIDITEM(X, Y, 0)
NEXT Y: NEXT X
LOCATE 25, 1: PRINT STRING$(80, " "); : LOCATE 25, 1
PRINT " F2 = New Game   F10 = Quit   Enter = Reveal
  Space = Mark; ";
LOCATE 25, 66: PRINT "Bombs left:" +
  STR$(NUMBOMBS);
LOCATE 18, 61: PRINT STRING$(12, 32);
LOCATE 19, 61: PRINT STRING$(2, 32) + "Time:" +
  STRING$(5, 32);
LOCATE 20, 61: PRINT STRING$(12, 32);
FIRSTMOVE = 1
END SUB

SUB SHOWGRIDITEM (X, Y, MODE)
    IF MODE = 1 THEN COLOR 3, 8 ELSE COLOR 8, 3
    SELECT CASE SHOW(X, Y)
    CASE 0
        LOCATE (Y - 1) * 2 + 3, (X - 1) * 4 + 12
        PRINT STRING$(3, 177);
    CASE 1
        LOCATE (Y - 1) * 2 + 3, (X - 1) * 4 + 12
        IF GRID(X, Y) = 9 THEN
            COLOR 12: PRINT CHR$(32) + "*" +
              CHR$(32);
        ELSEIF GRID(X, Y) > 0 THEN
            PRINT STR$(GRID(X, Y)) + CHR$(32);
        ELSE
            PRINT STRING$(3, 32);
        END IF
    CASE 2
        IF MODE = 1 THEN COLOR 3, 12 ELSE COLOR 12, 3
        LOCATE (Y - 1) * 2 + 3, (X - 1) * 4 + 12
        PRINT CHR$(32) + CHR$(2) + CHR$(32);
    CASE ELSE
    END SELECT
END SUB

SUB SHOWNEIGHBORS (X, Y)
COLOR 8, 3
IF GRID(X, Y) = 0 AND SHOW(X, Y) = 1 THEN
   IF Y > 1 AND X > 1 THEN
       X1 = X - 1: Y1 = Y - 1: GOSUB NEIGHBOR
   END IF
   IF Y > 1 THEN
       X1 = X: Y1 = Y - 1: GOSUB NEIGHBOR
   END IF
   IF Y > 1 AND X < 10 THEN
       X1 = X + 1: Y1 = Y - 1: GOSUB NEIGHBOR
   END IF
   IF X > 1 THEN
       X1 = X - 1: Y1 = Y: GOSUB NEIGHBOR
   END IF
   IF X < 10 THEN
       X1 = X + 1: Y1 = Y: GOSUB NEIGHBOR
   END IF
   IF Y < 10 AND X > 1 THEN
       X1 = X - 1: Y1 = Y + 1: GOSUB NEIGHBOR
   END IF
   IF Y < 10 THEN
       X1 = X: Y1 = Y + 1: GOSUB NEIGHBOR
   END IF
   IF Y < 10 AND X < 10 THEN
       X1 = X + 1: Y1 = Y + 1: GOSUB NEIGHBOR
   END IF
END IF
EXIT SUB
NEIGHBOR:
   IF SHOW(X1, Y1) = 0 THEN
       SHOW(X1, Y1) = 1: OPENX = OPENX + 1
       CALL SHOWGRIDITEM(X1, Y1, 0)
       IF GRID(X1, Y1) = 0 THEN CALL SHOWNEIGHBORS(X1, Y1)
   END IF
   RETURN
END SUB
```

End

But beware: The game lets you mark as many squares as you want. A counter in the bar at the bottom of the screen tells you the number of bombs remaining unmarked.

The counter starts at 15 and is decremented by one each time you mark a square, even if you mark it incorrectly. (If you realize you've made a mistake, you may unmark a square by highlighting it and pressing Enter again.) If you mark more than 15 squares, the counter displays a negative number.

If you're lucky (or smart) enough to uncover a square with no adjacent bombs, the program uncovers all eight surrounding squares automatically. If any of those new squares have no adjacent bombs, those squares are uncovered, as well. Sometimes, a lucky move will uncover half the grid, providing numerous clues to the locations of the remaining bombs.

To win the game, you must uncover all 85 empty squares. To up the pressure, a timer starts counting off seconds as soon as you uncover your first square. You'll know you're good when you can win a round of Mad Bomber in about 45 seconds. (If you're acquainted with Windows 3.1, you may recognize the similarities between the rules for Mad Bomber and those for Minesweeper, one of the free games that comes with Windows.)

Program Highlights

Although I don't have room to provide a line-by-line explanation of the program, I do want to point out a few key aspects of its operation.

I used two arrays to track the playing grid. The first, GRID, contains the contents of each grid square: A number from zero to 8 indicates the number of bombs adjacent to the square, and a value of 9 indicates that the square contains a bomb. The second array, SHOW, keeps track of whether you've uncovered or marked a square.

Both arrays are defined as 12-by-12 matrices, even though the game field is only a 10-by-10 matrix. The outer edge of each array (all cells in row zero and row 11 and all cells in column zero and column 11) is ignored. This design simplifies the logic of INITIALIZEGAME, particularly the section that counts the number of bombs adjacent to each cell within the playing field. Had I defined GRID as a 10-by-10 matrix, I would have needed additional boundary checks throughout that routine.

In addition to drawing the game board, the subprogram INITIALIZEGAME randomizes the location of the 15 bombs, then, based on bomb placement, calculates the number (zero to 8) to store in the other cells of the grid.

SHOWGRIDITEM displays a particular grid square. If the player uncovers the square, this routine displays the square's contents. If the player marks the cell, it displays the mark; otherwise, it displays a block to indicate that the player hasn't uncovered or marked the square. The third parameter passed to this subprogram, MODE, helps determine which color to display in the square. If the value of MODE is 1, the colors are inverted, which highlights the square.

The program's most interesting routine is SHOWNEIGHBORS, a subprogram called whenever the player uncovers a blank square. SHOWNEIGHBORS first uncovers all eight adjacent squares (assuming that the square doesn't lie on the edge of the grid). If one of the neighboring cells is also blank, the subprogram calls itself and uncovers another set of neighboring cells. This technique, called *recursion*, isn't used commonly in QBasic programs. But in this case, it's the most efficient way to uncover empty squares.

Making It Better

If you want to tinker with QBasic, Mad Bomber offers several do-it-yourself opportunities.

The first of these home-improvement projects—increasing the level of difficulty by increasing the number of bombs—is simple, because I used a CONST statement to establish the number of bombs. As noted above, the program as written hides 15 bombs. To increase the number of bombs, substitute a higher value in the tenth line. For example, if you change CONST to read as follows:

```
CONST NUMBOMBS = 20
```

the program hides 20 bombs within the playing field.

Were I not constrained by space, I'd make three other enhancements to Mad Bomber. First, I'd track high scores by writing outcomes to a file. Second, I'd consider letting the player vary the size of the grid. Finally, I'd investigate ways to eliminate the 1:15 chance of losing on the first move.

Keeping track of the best scores would mean storing the highest ones in a file, which entails adding statements to open the file, reading the previous best score, comparing it with the current player's score, and updating the file with new information if the player beats the previous best score.

In addition, you'd want to add a routine that displays a congratulatory message and asks for the player's name, which you'd show on screen beside the score.

Increasing the size of the playing field would involve extensive changes to the routines in INITIALIZEGAME that draw the board and randomize it, as well as modifications to the routines that move that little cursor around the screen.

Because each square on the grid requires three display lines and five display columns, a 10-by-10 matrix is about the largest game board you can create with the screen in 25-line mode. To create a larger board, you'd have to switch the display to 43-line mode.

And to eliminate the possibility of uncovering a bomb right off the bat, I suggest rewriting the program so that it randomizes the grid after the first move.

All these changes would improve the program, but none of them is absolutely essential—you'll still get plenty of hours of playing fun running Mad Bomber just as it is. ■

DOS TIP

THE SKY'S THE LIMIT

Receiving an "Unable to create directory entry" message while attempting to copy numerous small files from a hard disk to a floppy disk is frustrating, particularly when you know there's plenty of free space on the floppy disk. The problem arises because a typical floppy disk can accommodate only a limited number of entries in its root directory.

The solution is simple: Create a subdirectory on the floppy disk. Regardless of whether you're working with a hard disk or a floppy disk, subdirectories may contain an unlimited number of files.

To make a subdirectory on a floppy disk, follow the same procedure as on a hard disk. Log onto the floppy drive and type the following line:

```
MD dirname
```

Then use this command to switch into the subdirectory before beginning the copy operation:

```
CD dirname
```

The accompanying table, "Root-Directory Capacities," shows the maximum number of root-directory entries DOS allows on common floppy disks and hard disks.

—John Wolfskill

ROOT-DIRECTORY CAPACITIES

Disk Type	Capacity
single-sided 180K 5.25-inch floppy	64 files
double-sided 320K 5.25-inch floppy	64 files
double-sided 360K 5.25-inch floppy	112 files
double-sided 720K 3.5-inch floppy	112 files
double-sided 1.2MB 5.25-inch floppy	224 files
double-sided 1.44MB 3.5-inch floppy	224 files
hard disk (with 12- or 16-bit FAT)	512 files

REUSABLE SUBPROGRAMS

Sorting It All Out

Whether you work with letters or numbers, organizing those lists can be a massive job. Storing information in QBasic arrays is only half the battle—these two sort modules will rearrange your data and keep your files on track.

by Antonio and Alfonso De Pasquale

If you're in the habit of using QBasic arrays to store phone lists, personnel information, sales figures, and the like, it's a sure bet that someday you'll want to rearrange your data.

Maybe you'll need to list contacts by area code, or perhaps you'll want to rank employees according to salary. Whatever the reason, you probably won't have hours in which to write and debug a sort routine.

But if you add the two accompanying subprograms (contained in the listing SORT.BAS, page 64) to your library of reusable code, lack of time won't pose a problem.

You can decide whether your situation calls for a bubble sort or a shell sort, merge the appropriate subprogram with your existing QBasic program, add a few lines of code, and get right down to the business of sorting your arrays. (See the sidebar, "A Sort of a Different Color," for information on using DOS's SORT command to arrange your files.)

What Are Arrays?

Arrays are tools for storing, organizing, and processing data efficiently in a computer's memory. They're particularly handy for holding data items of the same type and purpose.

For example, a calendar program might use arrays to hold the names of the days of the week, the number of days in each month, and so on. From a programming standpoint, it makes more sense to store the days of the week in a single array than to declare a different variable for each day of the week. Creating an array involves less coding and uses the computer's memory more efficiently.

QBasic uses a DIM (for *dimension*) statement to declare arrays. The basic syntax is the key word, DIM, followed by a parenthetical phrase that provides the array's name and size. The following commands, for instance, declare two arrays: a string array consisting of seven elements and a numeric array consisting of 31 elements:

```
DIM DAY$(7)
DIM MONTHS(31)
```

QBasic lets you declare either numeric or string arrays. As the following example shows, you may also use variables instead of numbers when declaring the size of an array:

```
ARSIZE=250
DIM TOTAL(ARSIZE)
```

The first line sets the variable ARSIZE to 250; the second sets, or *dimensions*, the array to the size specified by ARSIZE.

Bubble or Shell?

The simplest method of arranging the elements of a QBasic array is a *bubble sort*. The procedure involves looking at adjacent elements in the array and swapping their values when the elements are out of order. When you're sorting values in ascending order, swapping values moves the largest values down the array, permitting the smallest values to "bubble" to the top.

To sort data properly, the algorithm for a bubble sort must make

---

Free-lance writers and programmers Antonio and Alfonso De Pasquale are frequent contributors to DOS Resource Guide.

several passes through an array. The exact number of passes required is the total number of elements in the array minus one. If your array contains 500 data items, for instance, a bubble sort requires 499 passes to sort it properly.

As you might expect, a bubble sort is best suited for smaller arrays. The algorithm, although easy to follow, is inefficient when working with large arrays, because each element moves up or down by only one position in each pass.

Named after its creator, Donald Shell, a *shell sort* provides a more efficient method of processing larger arrays. Whereas a bubble sort compares and swaps adjacent elements, the shell sort compares and swaps nonadjacent elements. It begins by comparing the first element of the array with the middle element, referred to as the *gap*. If no swap is made, the gap is halved and nonadjacent elements are compared again. The gap is compared and halved repeatedly until a swap is made.

Because this algorithm requires fewer comparisons in each pass through the array, a shell sort arranges data more quickly than a bubble sort. Consequently, a shell sort is superior to a bubble sort in most circumstances. The bubble sort's only real advantages are that the routine is compact and its logic is easy to follow. When working with moderate to large arrays, the shell sort should be your routine of choice.

Entering the Program

The accompanying program listing, SORT.BAS, shows how you might write subprograms to perform these two types of sorts. The first half of the program prompts you for an array size, generates an array of random numbers, and asks whether you want to perform a bubble or a shell sort. The program then displays the starting and ending times of the sort and displays the results on screen. This part of the listing is included strictly for illustrative purposes. The meat of the listing is the pair of subprograms in the second half.

To create SORT.BAS, load QBasic and type in the first part of the listing. (Remarks inserted in the program indicate the end of this section.) When you finish, save the file to disk by selecting the *Save* option from the *File* menu and typing `SORT.BAS` as the filename.

To create the first subprogram and link it to SORT.BAS, select the *New Subs* option from the *Edit* menu and name the subprogram BUBBLESORT. (Be sure that SORT.BAS is loaded before typing in the subprogram.) QBasic fills in the first part of the first line of the subprogram, as well as the last line. You must type everything in between. The next time you save the main program, QBasic will save BUBBLESORT along with it, adding the proper

A SORT OF A DIFFERENT COLOR

The sort routines provided in the accompanying article let you sort QBasic arrays. If you work in DOS, you may use its SORT command to arrange the contents of a file and send the results to another file, the screen, or a device such as a printer. Please bear in mind, however, that SORT provides very different services from those offered by our QBasic sorting procedures. The SORT command sorts the contents of a file; BUBBLESORT and SHELLSORT sort the contents of arrays.

The basic syntax of the SORT command is as follows:

```
SORT inputfile > outputfile
```

If you have a file of names called NAMES.TXT, for example, the following command sorts the file and stores the results in a file called NAMES2.TXT.

```
SORT NAMES.TXT > NAMES2.TXT
```

By default, SORT arranges data in ascending order. To sort a file in descending order, use the /R switch (for *reverse*:

```
SORT /R NAMES.TXT > NAMES2.TXT
```

You may also sort tabular data by column with the /+N switch, substituting for *N* the number of the column on which to sort. For instance, if you want to sort columnar data by last names, which are in column 15, the proper command is this:

```
SORT /+15 NAMES.TXT > NAMES2.TXT
```

To send the results of the sort to the screen, simply omit the greater-than sign (>) and the name of the output file, as in the following example:

```
SORT NAMES.TXT
```

Finally, the following command sends the results of the sort to your printer:

```
SORT NAMES.TXT > PRN
```

(*PRN* is the device DOS associates with your printer's parallel port.)

—A.D. and A.D.

DECLARE statement at the top of the main program.

After typing in BUBBLESORT, create the QBasic subprogram SHELLSORT in exactly the same way: Select *New Subs* from the *Edit* menu and begin typing.

When you finish, don't forget to save the program.

Using the Program

To run the program, select *Start* from the *Run* menu or press Shift-F5. You also may run the program from DOS by typing `QBASIC/RUN SORT.BAS`. The screen will clear, and a greeting will appear. The program will then take a few moments to create an array of 250 randomly generated numbers. When the menu appears, you have three

SORT.BAS, a demonstration QBasic program, includes two subprograms: one that performs bubble sorts and another that performs shell sorts.

```
' Antonio & Alfonso De Pasquale

'  SORT.BAS

DECLARE SUB BUBBLESORT (ARRAY())
DECLARE SUB SHELLSORT (ARRAY())

MAIN:
    CLS : CLEAR : RANDOMIZE TIMER: ENTRIES = 0
    PRINT TAB(32); "Sort Demonstrator"
    PRINT TAB(25); "By Antonio & Alfonso
        De Pasquale"
    FOR x = 1 TO 80: PRINT CHR$(196); : NEXT x:
        PRINT : PRINT
    PRINT "This program demonstrates
        the differences in speed between"
    PRINT "a bubble sort and a shell sort.": PRINT

    INPUT "Please enter an array size
        (50 - 500) ", ENTRIES
    IF ENTRIES = 0 THEN CLS : END
    IF ENTRIES < 50 OR ENTRIES > 500
        THEN GOTO MAIN
    DIM NUMLIST(ENTRIES): PRINT

    PRINT "Please wait a moment while the program
        creates an array of "; ENTRIES;
        " random numbers..."
    FOR X = 1 TO ENTRIES: NUMLIST(X) = INT(1000 *
        RND + 1): NEXT X
    PRINT : PRINT "Please select one:  (B)ubble
        Sort, (S)hell Sort, (Q)uit Program: ";

    K$ = ""
    DO WHILE K$ <> "B" AND K$ <> "S" AND K$ <> "Q"
        K$ = UCASE$(INKEY$)
    LOOP

    PRINT K$: IF K$ = "Q" THEN CLS : END
    PRINT : PRINT "The sort was started at   ";
        TIME$

    IF K$ = "B" THEN
        BUBBLESORT NUMLIST()
    ELSE
        SHELLSORT NUMLIST()
    END IF

    PRINT "The sort was completed at "; TIME$
    PRINT : PRINT "Press Enter to view
        the sorted numbers"
    PRINT : DO: LOOP UNTIL INKEY$ = CHR$(13)
    CLS : PRINT "Here are the sorted numbers...":
        PRINT
    FOR x = 1 TO ENTRIES: PRINT USING "  ###";
        NUMLIST(X); : NEXT X
    PRINT : PRINT : PRINT "Press Enter
        to continue"
    DO: LOOP UNTIL INKEY$ = CHR$(13)
    GOTO MAIN

' This marks the end of the main program.
' The reusable subprograms begin here.
'

SUB BUBBLESORT (ARRAY())

LOW = LBOUND(ARRAY)
HIGH = UBOUND(ARRAY)
SWITCH = 0

DO WHILE SWITCH = 0
    SWITCH = 1
    FOR INDEX = (LOW + 1) TO (HIGH - 1)
        IF ARRAY(INDEX) > ARRAY(INDEX + 1) THEN
            SWAP ARRAY(INDEX), ARRAY(INDEX + 1)
            SWITCH = 0
        END IF
    NEXT INDEX
LOOP

END SUB

SUB SHELLSORT (ARRAY())

LOW = LBOUND(ARRAY)
HIGH = UBOUND(ARRAY)
GAP = INT(HIGH / 2)

DO WHILE GAP <> 0
    SWITCH = 0
    DO WHILE SWITCH = 0
        SWITCH = 1
        FOR INDEX = (LOW + 1) TO (HIGH - GAP)
            IF ARRAY(INDEX) > ARRAY(INDEX + GAP) THEN
                SWAP ARRAY(INDEX), ARRAY(INDEX + GAP)
                SWITCH = 0
            END IF
        NEXT INDEX
    LOOP
    GAP = INT(GAP / 2)
LOOP

END SUB
```

End

choices: pressing *B* to perform a bubble sort, *S* to perform a shell sort, or *Q* to quit the program.

When you press *B* or *S*, the program displays the starting time and begins the sort. When sorting is complete, the program displays the ending time. Press Enter to view the sorted numbers on screen. As we mentioned earlier, a bubble sort is best suited for small arrays; a shell sort dramatically improves performance with larger arrays.

Creating a Sort Library

To use the two sorting subprograms in your own programs, you must first isolate them from the original program and then add them to your code. Start by using your mouse or the Shift keys in conjunction with the arrow keys to highlight the sections of SORT.BAS that you don't need and delete them. Then, use the File menu's Save As option to save the subprograms under a different name, perhaps SORT.LIB. (You may save each subprogram separately, of course.)

Unfortunately, QBasic doesn't provide an import feature for merging subprograms into a loaded program. You may use a DOS COPY command such as the following, however, to append the subprograms to your own QBasic code:

```
COPY filename + SORT.LIB
```

Just substitute the name of your QBasic program for *filename*. This command attaches the two subprograms to the bottom of your QBasic program. The next time you call up QBasic and load your program, the proper DECLARE statements will be added to your code, and the subprograms BUBBLESORT and SHELLSORT will appear in the QBasic *Subs* list.

Calling the Functions

Whether you want to perform a bubble sort or a shell sort, you must pass the name of the array to be sorted to the subprogram. The following QBasic code illustrates the proper syntax for calling the subprograms:

```
BUBBLESORT ARRAY1()
SHELLSORT  ARRAY2()
```

In the first line, ARRAY1() is the name of the array; notice that you must include the parentheses to inform QBasic that this variable refers to an array. The subprogram then takes over and sorts the data in the stated array. In the second line, ARRAY2() is the name of the array. In your own programs, you may substitute any valid numeric array name for ARRAY1() and ARRAY2().

Further Modifications

As written, the two sort subprograms are designed to arrange numeric data, but you may modify them to work with string data (alphabetic characters), as well.

To do this, simply add a dollar sign ($) to the array name when defining, declaring, or calling the subprograms. The dollar sign here informs QBasic that you're working with a string array rather than a numeric array. Your DECLARE statements, for example, would look like this:

```
DECLARE SUB BUBBLESORT (ARRAY$())
DECLARE SUB SHELLSORT (ARRAY$())
```

The first lines of the functions themselves would look like this:

```
SUB BUBBLESORT (ARRAY$())
.
.}more work done in these lines
.

SUB SHELLSORT (ARRAY$())
.
.}more work done in these lines
.
```

The code to call these routines would look like this:

```
BUBBLESORT ARRAY1$()
SHELLSORT  ARRAY2$()
```

Those are the basics of working with arrays. Even if your QBasic programs don't require sort routines right now, you might want to spend a little time experimenting to make sure you grasp the fundamental rules. Then just tuck the subprograms into your QBasic library for future reference. ■

DOS TIP

A WELL-KEPT PATH

If you have trouble keeping your AUTOEXEC.BAT's PATH statement within DOS's 127-character limit, maybe this trick will help. Eliminate all occurrences of C: from the PATH statement, changing a statement such as this one:

```
PATH=C:\;C:\DOS;C:\WINDOWS;C:\BATCH;C:\SYSTEM;C:\TEMP
```

to read:

```
PATH=\;\DOS;\WINDOWS;\BATCH;\SYSTEM;\TEMP
```

This works because DOS interprets the backslash (\) to mean *begin with the root directory of the current drive.*

Before you try this, however, first save your old version of AUTOEXEC.BAT to a file named AUTOEXEC.OLD, or something similar. That way, you can easily reinstate the old version at a later date. And remember, if you log onto another drive, DOS will look for the programs on that drive, because you've omitted all references to a drive letter. You should make this change only if you always work from drive C.

—Lane Olinghouse

DOS RESOURCE GUIDE

Readers' Queue

Wipe It Out

I often want to delete files so that they can't be recovered using utilities such as the MS-DOS UNDELETE command. To do this, I developed a QBasic program called QWIPE.BAS; it not only deletes files but also overwrites the files' contents. Although QWIPE.BAS isn't as fancy as The Norton Utilities' WipeFile program and other similar commercial software, it works well.

The program starts by asking you the name of the file you want to wipe out; it then overwrites that file in 2K blocks, using zeros, 1s, and a random character. Finally, QWIPE.BAS changes the name of the overwritten file to a single letter, which makes it virtually impossible for a nosy user to utilize the filename to determine what the file contained originally.

You may customize the program in several ways. You may change the characters used to overwrite the file, for one thing. To speed the file-wiping process, you may also reduce the number of times the file is overwritten to fewer than three.

Because my hard disk is organized into 2K clusters, I set the block size to 2048 bytes (2K). If your hard drive is large, and you've partitioned it as one drive, the cluster size may be 4K or larger. You should change the size of the block to match your hard disk's. (To find out the block size, invoke CHKDSK and read the line that tells the number of bytes in each allocation unit.)

Finally, you may modify the program to read the file after the last write is carried out, which lets you verify that the file contains only the overwrite characters. If you do this, you should also add an error message that gives you the option of wiping the file again.

Vincent D. O'Connor
Babitt, Minn.

The Power of One

A couple of years ago, I went into a local department store that sells computers; each system was running a lame demo program. As a joke, I stopped the demo on one machine (when the salesperson wasn't looking) and typed in a demo of my own. It was a simple graphics routine I had worked out: a series of gray 3-D spheres that sprouted at random screen locations. The spheres looked like globs, so I called the program GLOBS.BAS. Because the program fits on one line, I typed it into QBasic's Immediate window, pressed Enter, and sidled away.

Ever since then, writing one-liners has been one of my hobbies. I've even gotten a few friends involved. We have a few basic rules:

- Each program must fit into one line of QBasic code.
- You must be able to run the one-liner in the Immediate window.
- It must exploit graphics or sound (or both) in interesting ways.
- Its output must make other users think that the underlying code is complex.

I've enclosed three of my favorites to share with *DRG* readers: the original GLOBS.BAS plus INYRFACE.BAS and BUBBLES.BAS.

Tom Sales
Somerset, N.J.

GLOBS.BAS is a one-line QBasic program that displays ever-changing monochrome "blobs" on your screen.

```
SCREEN 13: DO: X = RND * 320: Y = RND * 240: C = 31: R = RND * 25
  + 10: CS = C / R / 2.3: FOR I = 1 TO R: CIRCLE (X, Y), I, C: C
  = C - CS: NEXT: LOOP
```

End

INYRFACE.BAS displays a series of colored bubbles that grow steadily in size as the program works its way through a series of STEP increases. When the program reaches the maximum value, it loops back to the beginning.

```
SCREEN 12: DO: CLS : FOR I = 200 TO 2 STEP -1: FOR R = 1 TO 400 /
  I: CIRCLE (RND * 640, RND * 480), RND * R, RND * 16 + 1: NEXT:
  NEXT: LOOP
```

End

BUBBLES.BAS displays a changing array of colored circles that bubble toward the top of your screen, with audio accompaniment.

```
SCREEN 12: DO: R = RND * 30 + 3: CIRCLE (RND * 640, 430), R, RND *
  16 + 1: LOCATE 30, 1: PRINT : SOUND R * 50, 1: LOOP
```

End

Readers, Tom Sales and company have thrown down the gauntlet. If you're up to accepting the challenge, we're up for publishing some of the better one-line QBasic submissions we receive. —*Eds.*

IMPROVEMENTS & CORRECTIONS

Password Validation

As you pointed out, my original version of QBCRYPT.BAS ["The Right to Privacy" in "Reader's Queue," *DRG*#15, the May 1994 issue, page 63] couldn't detect whether the wrong password was entered.

I have a simple fix; however, it works only if you don't use extended ASCII characters (those with values higher than 127) in your text files.

First, add the following code to the end of line 59 of the program (the line after IF X < 1 THEN X = ((X + 255) – 32)):

```
IF X > 127 THEN GOTO WRONGKEY
```

Then add these lines to the end of the program:

```
WRONGKEY:
  LOCATE 15, 1: PRINT "WARNING
    INCORRECT PASSWORD !!!"
  PRINT "DECIPHERING PROCESS
    ABORTED !!"
  FOR DLAY = 1 TO 5000: NEXT DLAY
  CLOSE
  KILL NAME$
  GOTO BEGIN
```

Chris Stoddard
APO AE 09267

QWIPE.BAS lets you overwrite a file so that its contents can't be recovered using UNDELETE or a similar utility.

```
DEFINT A-Z

DECLARE SUB WORKING (KOUNT!)

TYPE WIPECHAR  'The size of the string that
   WIPE AS STRING * 2048  'overwrites the file
     to be wiped.
END TYPE
DIM CHAR AS WIPECHAR

ON ERROR GOTO ERRTRAP
START:
CLS
LOCATE 12, 1
LINE INPUT "Type the name of the file to be
  wiped: "; FILE$
IF FILE$ = "" THEN
   CLS : END
END IF
OPEN FILE$ FOR INPUT AS #1
CLOSE : LOCATE 14, 1: BEEP
PRINT "Wipe this file? (Y/N) ";
YN$ = UCASE$(INPUT$(1))
IF YN$ = "N" THEN
   FILE$ = "": GOTO START
END IF
CLS : LOCATE 12, 1
PRINT "Wiping file..."
OPEN FILE$ FOR RANDOM AS #1 LEN = 2048
SIZE = LOF(1) \ 2048  'Determine the number of
  string
IF SIZE < 1 THEN SIZE = 1  'sections to use to
  overwrite file.
FOR J = 1 TO 3  'Write three times to ensure that
                'the file is overwritten.
   FOR K = 1 TO 2
      IF K = 1 THEN CHAR.WIPE = STRING$(2048,
        "0") 'Writes zeros over file.
      IF K = 2 THEN CHAR.WIPE = STRING$(2048,
        "1") 'Writes 1s over file.
      FOR I = 1 TO SIZE
         PUT #1, I, CHAR.WIPE: WORKING (I!)
      NEXT I
   NEXT K
NEXT J
RANDOMIZE TIMER
RANDCHAR = INT(RND * 256)  'Make random
  character.
CHAR.WIPE = STRING$(2048, RANDCHAR)
FOR I = 1 TO SIZE        'Write random character
   PUT #1, I, CHAR.WIPE: WORKING (I!) 'over file.
NEXT I
CLOSE : NAME FILE$ AS "$": KILL "$"  'Make
  filename one character.
CLS : LOCATE 12, 1: BEEP
PRINT "File wiped! Press any key to continue..."
DO: LOOP WHILE INKEY$ = ""
GOTO START

ERRTRAP:
IF ERR <> 53 THEN
   PRINT ERR: CLOSE : END
ELSE
   BEEP: LOCATE 20, 1: PRINT "Error!
     File not found!"
   LOCATE 21, 1: PRINT "Press any key
     to continue..."
   DO: LOOP WHILE INKEY$ = ""
   FILE$ = ""
   RESUME START
END IF

SUB WORKING (KOUNT!)
   LOCATE 12, 15
   IF KOUNT! / 2 = KOUNT! \ 2 THEN
      PRINT "/"
   ELSE
      PRINT "\"
   END IF
END SUB
```

End

START-UP CLINIC

The Power Of Some

by Jack Nimersheim

Is your 286 trolling along at 10MHz, while your neighbor's runs like something that belongs in Siegfried & Roy's new act? Hurts, huh?

Many computer owners suffer from the same malady: power envy. If the price of computing hasn't fallen far enough to meet your budget, you still have a couple of alternatives. You might consider trading your equipment (fat chance!), swapping the CPU for a faster chip, or replacing the motherboard—that is, swap not only the chip, but the bus and ports, as well. Some readers want to know where to begin.

> *Many computer owners suffer from the same malady: power envy.*

Trade-In Value

I'd like to convert my IBM 286 to a faster system. Stan Miastkowski's "Hardware Update" column in DRG*#13 [January 1994, page 59] discussed processor upgrades. But what's the difference between that and installing a new motherboard in an old computer?*

Elmer Albeluhn
Port Alice, British Columbia
Canada

Swapping your processor or CPU chip is much different from replacing your motherboard. Think about your car. Consider some of your options when your automobile begins to show its age.

For instance, suppose a mechanic tells you that your engine's valves are wearing out. You could replace them, something like upgrading your computer's central processing unit (CPU), and that would improve the situation—maybe bump up your car's performance by a couple of miles per gallon. On the downside, however, you'd still be saddled with an old, probably less efficient, engine.

By replacing your computer's chip—even if you've chosen one of the faster 486DX chips—you're still running a less efficient computer: Its bus speeds are slower, its memory expansion may be limited, and its BIOS (basic input/output system) routines are archaic. For example, your 80286 motherboard probably contains 16-bit expansion slots, rather than the faster 32-bit slots on motherboards built for 80386 and 80486 computers.

Even more critical, the BIOS chips in many AT computers won't recognize or support newer, high-capacity hard disks—an essential component for using today's storage-hungry applications. Unless you plan to keep your old hard drive, you'll also need to include a new BIOS as part of your upgrade plans.

A second approach to upgrading older equipment is to replace your 80286 motherboard with one designed to take full advantage of today's PC hardware. In other words, buy a whole new engine. A replacement motherboard should include, among other things, a BIOS that recognizes faster memory, larger hard disks, higher-resolution video, and so on.

One caveat: Motherboards come in many different shapes and sizes. Make sure your new one matches the dimensions of the one you're replacing. If it doesn't, you'll face the unenviable task of modifying either the new board or your computer case.

A third alternative remains, however: buying a new car. Unlike the escalating sticker prices of today's automobiles, the cost of computers is plummeting—to under $1000. The biggest advantage is that every component offers improved performance over its counterpart in your current system—and your new purchase will also be covered by a warranty.

Color Me Blue

Recently, while experimenting with DOSSHELL switches (/T for text-only, /G for graphics, and so on), I managed to lose all the color options—probably when I included /B to specify a black-and-white display. Now I can't restore color to my DOS Shell. I've tried the Color command in DOSSHELL's Options menu, but it doesn't work anymore. What's happening?

Stanley F. Goldfein
New York, N.Y.

Your problem has two possible causes and solutions. One situation is simple to fix; the other requires a bit more effort. Let's begin with the easier one.

Are you including the /B switch each time you run DOSSHELL? For example, have you added it to a batch file you use to start DOSSHELL? If so, that's your problem. Specifying the /B switch overrides DOSSHELL's color capabilities completely. Even if you use the Color option subsequently to select a color scheme, your display will remain in black-and-white mode. To restore color capabilities to DOSSHELL, remove the /B switch from your batch file.

The second possibility is that your DOSSHELL.INI file has been corrupted somehow. This file, which usually resides in the DOS directory, contains information DOSSHELL needs to manage its operations properly, including the various hues associated with a given color scheme. For example, the following section from DOSSHELL.INI lists the settings your system uses to define the background colors for DOSSHELL's "Emerald City" scheme:

```
background =
base = brightwhite
highlight = brightgreen
selection = green
alert = brightwhite
menubar = brightwhite
menu = brightwhite
disabled = brightwhite
accelerator = brightwhite
dialog = brightwhite
button = green
elevator = white
titlebar = green
scrollbar = black
borders = black
drivebox = brightwhite
driveicon = brightwhite
cursor = brightcyan
```

The "brightgreen," "green," and "brightcyan" settings in this section indicate some of the colors that appear in the "Emerald City" DOSSHELL display. Still other color schemes included in DOSSHELL.INI use different settings to define additional display options.

DOSSHELL.INI is a text file, so you can use the DOS EDIT.COM utility to view and modify its contents. To do that, switch to your DOS directory, type `EDIT DOSSHELL.INI`, and press Enter. This sequence starts DOS's Editor and loads the DOSSHELL.INI file into memory. (One suggestion: Make a backup copy of your original DOSSHELL.INI file before attempting to edit it, just in case your modifications cause subsequent problems starting the DOS Shell.)

Press the PgDn key to scroll through DOSSHELL.INI until you reach a section that begins with the following line:

```
color =
```

This portion designates the color combinations that make up the range of display modes available to the DOS Shell. Once you locate this section, you can determine which screen elements are assigned which colors in various modes. For example, the "Basic Blue" color scheme uses "brightred" for its foreground alert setting, "cyan" for the accelerator setting, and "blue" for background highlights. All other screen elements in "Basic Blue" use variations on black and white settings. In truth, none of the DOS Shell color schemes is very elaborate. As a rule, they rely on only two or three actual colors for emphasis, using black, white, and gray shading for major screen elements.

Admittedly, the screen-element and color names used in DOSSHELL.INI tend to be somewhat cryptic. With some experimentation, however, you can probably unravel the file's mysteries. (And remember, you always have that backup copy of your original DOSSHELL.INI file, should problems arise.) If you do uncover what appears to be a potential problem, try correcting it, and then use the Editor's File/Save command to save DOSSHELL.INI back to disk. Your changes will take effect the next time you start the DOS Shell.

If neither approach works, try reinstalling DOS from your original distribution disks. DOS's Setup utility creates the DOSSHELL.INI file as part of its installation routine, which corrects the problem automatically. ■

Jack Nimersheim is the author of more than 20 books on computer-related topics, including Slick Tricks: MS-DOS *(Random House Electronic Publishing).*

DOS RESOURCE GUIDE

Q&A

DOS

Q I'm unclear about how the DOS PATH command, which I place in my AUTOEXEC.BAT file, actually works. For example, when I'm working in a directory named UTIL and need to start a program in a directory I've named LETTERS, what difference does it make whether the LETTERS directory is mentioned in my PATH statement or not? I've tried the above example and fail to see a difference in my computer's speed or performance when DOS attempts to locate and start a program.

A The purpose of listing selected DOS directories in your AUTOEXEC.BAT file's PATH statement isn't to speed up the operation of the DOS file system. PATH is designed to save you from typing unnecessary keystrokes. For example, suppose you're working in C:\UTIL and want to start a program named WRITER.EXE, which is located in C:\LETTERS. If you fail to include C:\LETTERS in your PATH statement, DOS won't be able to find the program when you type the following line:

```
C:\UTIL>WRITER
```

Instead, DOS will greet you with the error message "Bad command or filename" because it has no clue as to where the program may be located. In this case, there are two ways you can start WRITER.EXE. Unfortunately, both methods do require you to type substantially more keystrokes.

First, you can log the directory where WRITER.EXE is located by typing this line:

```
C:\UTIL>CD\LETTERS
```

After you reach the LETTERS directory you must then type this line:

```
C:\LETTERS>WRITER
```

to start the program.

The second method requires you to type the complete DOS pathname of the program you want to start. For example:

```
C:\UTIL>C:\LETTERS\WRITER
```

However, after you add the directory LETTERS to your AUTOEXEC.BAT file's PATH statement, as follows:

```
PATH=C:\LETTERS
```

you can start WRITER.EXE from any directory in your system by typing this line:

```
C:\UTIL>WRITER
```

There are a few caveats and limitations you should know about the DOS PATH statement. First, the PATH search works only with executable files (those with the three-character file extension COM, EXE, or BAT). That means you can't use the PATH to locate data or text files. Second, DOS effectively limits the length of the PATH statement to 127 characters. If your hard drive contains many directories with executable files, you may need to be selective about the directory names you include in your PATH.

Q For the past six months I've been teaching myself to program in QBasic, the programming language provided with DOS 5. I've written a few programs I use now at work, and I'd like to incorporate them into a menu for others to use without having to teach each person how to start and run the QBasic editor. Is there a software utility I can use to turn my QBasic programs into executable files that can be started from the DOS prompt? Can you recommend software for Basic programming that will let me create executable files?

A What you need is a Basic compiler—a program that lets you create stand-alone (executable) files, using the BAS source files you've created in QBasic. I recommend Microsoft's QuickBasic 4.5. (See the box below.) You can buy it at most software stores or through mail-order software houses. QuickBasic is 100 percent compatible with your existing source files, which means you'll need to make only small modifications, if any, in your existing source files to create executable programs. In addition, QuickBasic comes with an enhanced program editor that's similar in function to the one provided with QBasic.

PRODUCT INFORMATION

QuickBasic 4.5, $199
Microsoft Corp.
1 Microsoft Way
Redmond, WA 98052-6399
800-426-9400
206-882-8080

Q Each time I attempt to start my computer game (called FX17.EXE), the program returns the error message "Error 53. Can't find configuration file FX17.DAT." Although I've included the program's DOS directory, called C:\GAMES, in my AUTOEXEC.BAT file's PATH statement, the game refuses to start unless I actually log the directory where the program is located. What gives?

A When you use the PATH statement to access a program from another directory, DOS doesn't actually shift to the directory where your program is stored.

Instead, DOS simply runs the program from the current directory. This can cause problems with programs that assume their associated data files are always located in the current directory.

One solution to this problem is to create a small DOS batch file that changes to the directory where your game is located, starts your game, then returns you to the root directory after the game ends.

For example, you can use a word-processing program to create the following DOS batch file named FX17.BAT:

```
@ECHO OFF
C:
CD\GAMES
FX17
CD\
```

You can use this method to create similar batch files for each program you want to run. Create a new directory called C:\BATCH in which to store the batch files:

```
C:>MD BATCH
```

Then add C:\BATCH to the end of your existing AUTOEXEC.BAT file's PATH statement:

```
PATH=C:\;C:\BATCH
```

Finally, reboot your PC so that DOS will recognize the change you've made in your PATH statement. Now you can start your programs and run them correctly from any directory on your hard drive.

HARDWARE

Q My 80286 PC is equipped with two floppy drives. (The A: drive is a 1.2MB 5.25-inch drive, and the B: drive is a 1.44MB 3.5-inch drive.) Recently drive A has lost its ability to read disks. My neighbor says the drive is probably running too fast, and if my hard drive ever fails at start-up I'll no longer be able to start my computer. Is this true? Since drive B works fine, is there a command I can use to tell my PC to look for a boot disk in drive B instead of drive A?

A Your neighbor is correct about the danger posed by your malfunctioning floppy drive. That problem, however, is probably unrelated to drive-spindle rotation speed.

The most likely cause is read/write-head misalignment. Although you can take your computer to a qualified technician to have the heads realigned, it's usually less expensive to replace the offending floppy drive with a new one (around $45 to $50).

On most PCs, you'll find that the start-up sequence routines, located in a set of permanent memory chips called the ROM BIOS, check automatically for the presence of a boot disk in drive A before telling the computer to start DOS from the start-up routines stored on your hard-disk drive. That lets you gain access to your computer should your hard-disk drive fail during the start-up process.

Many (but not all) PCs provide a Setup program that may let you select start-up options, such as designating the default start-up drive as drive B. In most cases, you can access Setup by pressing a key in response to a prompt that appears on screen in the early stages of system start-up.

If your own PC's Setup program doesn't provide this option, I suggest you repair or replace the malfunctioning floppy-disk drive as soon as possible.

Q Ever since I installed DOS 6 on my 33MHz 80486, I've had a problem with my system losing its CMOS hard-drive settings. Sometimes the settings are among the missing when I start my computer. Is this a problem with DOS, or is my hard drive a goner?

A If you haven't already done so, start by replacing your computer's CMOS storage battery, which may be losing power. Any computer dealer can sell you a replacement battery and help you install it.

I've also heard of isolated cases in which the symptoms you describe were caused by using EMM386.EXE with its HIGHSCAN feature. If you use EMM386.EXE, you may want to try removing the HIGHSCAN option from the line in your CONFIG.SYS file that loads this device driver, and then run MEMMAKER.EXE in custom mode to reconfigure your system.

Choose "No" in response to the program's "Scan Upper Memory Aggressively" option. ■

"Q&A" is compiled by Technical Editor John Wolfskill, who is president of Powertrain Development, a New Hampshire-based software-development company.

DOS RESOURCE GUIDE

DOS Resource Guide is a publication of International Data Group, the world's largest publisher of computer-related information and the leading global provider of information services on information technology. International Data Group publishes over 194 computer publications in 62 countries. Forty million people read one or more International Data Group publications each month. International Data Group's publications include: ARGENTINA's *Computerworld Argentina, Infoworld Argentina*; ASIA's *Computerworld Hong Kong, PC World Hong Kong, Computerworld Southeast Asia, PC World Singapore, Computerworld Malaysia, PC World Malaysia*; AUSTRALIA's *Computerworld Australia, Australian PC World, Australian Macworld, Network World, Mobile Business Australia, Reseller, IDG Sources*; AUSTRIA's *Computerwelt Oesterreich, PC Test*; BRAZIL's *Computerworld, Gamepro, Game Power, Mundo IBM, Mundo Unix, PC World, Super Game*; BELGIUM's *Data News (CW)*; BULGARIA's *Computerworld Bulgaria, Ediworld, PC & Mac World Bulgaria, Network World Bulgaria*; CANADA's *CIO Canada, Computerworld Canada, Graduate Computerworld, InfoCanada, Network World Canada*; CHILE's *Computerworld Chile, Informatica*; COLOMBIA's *Computerworld Colombia*; CZECH REPUBLIC's *Computerworld, Elektronika, PC World*; DENMARK's *CAD/CAM WORLD, Communications World, Computerworld Danmark, LOTUS World, Macintosh Produktkatalog, Macworld Danmark, PC World Danmark, PC World Produktguide, Windows World*; ECUADOR's *PC World Ecuador*; EGYPT's *Computerworld (CW) Middle East, PC World Middle East*; FINLAND's *MikroPC, Tietoviikko, Tietoverkko*; FRANCE's *Distributique, GOLDEN MAC, InfoPC, Languages & Systems, Le Guide du Monde Informatique, Le Monde Informatique, Telecoms & Reseaux*; GERMANY's *Computerwoche, Computerwoche Focus, Computerwoche Extra, Computerwoche Karriere, Information Management, Macwelt, Netzwelt, PC Welt, PC Woche, Publish, Unit*; GREECE's *Infoworld, PC Games*; HUNGARY's *Computerworld SZT, PC World*; INDIA's *Computers & Communications*; IRELAND's *Computerscope*; ISRAEL's *Computerworld Israel, PC World Israel*; ITALY's *Computerworld Italia, Lotus Magazine, Macworld Italia, Networking Italia, PC Shopping Italy, PC World Italia*; JAPAN's *Computerworld Today, Information Systems World, Macworld Japan, Nikkei Personal Computing, SunWorld Japan, Windows World*; KENYA's *East African Computer News*; KOREA's *Computerworld Korea, Macworld Korea, PC World Korea*; MEXICO's *Compu Edicion, Compu Manufactura, Computacion/Punto de Venta, Computerworld Mexico, MacWorld, Mundo Unix, PC World, Windows*; THE NETHERLANDS' *Computer! Totaal, Computable (CW), LAN Magazine, MacWorld, Totaal "Windows"*; NEW ZEALAND's *Computer Listings, Computerworld New Zealand, New Zealand PC World*; NIGERIA's *PC World Africa*; NORWAY's *Computerworld Norge, C/World, Lotusworld Norge, Macworld Norge, Networld, PC World Ekspress, PC World Norge, PC World's Produktguide, Publish & Multimedia World, Student Data, Unix World, Windowsworld; IDG Direct Response*; PANAMA's *PC World Panama*; PERU's *Computerworld Peru, PC World*; PEOPLE'S REPUBLIC OF CHINA's *China Computerworld, China Infoworld, PC World China, Electronics International, Electronic Product World, China Network World*; IDG HIGH TECH BEIJING's *New Product World*; IDG SHENZHEN's *Computer News Digest*; THE PHILIPPINES' *Computerworld Philippines, PC Digest (PCW)*; POLAND's *Computerworld Poland, PC World/Komputer*; PORTUGAL's *Cerebro/PC World, Correio Informatico/Computerworld, MacIn*; ROMANIA's *Computerworld, PC World*; RUSSIA'S *Computerworld-Moscow, Mir-PC, Sety*; SLOVENIA's *Monitor Magazine*; SOUTH AFRICA's *Computer Mail (CIO), Computing S.A., Network World S.A.*; SPAIN's *AmigaWorld, Computerworld Espana, Communicaciones World, Macworld Espana, NeXTWORLD, Super Juegos Magazine (GamePro), PC World Espana, Publish, Sunworld*; SWEDEN's *Attack, ComputerSweden, Corporate Computing, Lokala Natverk/LAN, Lotus World, MAC&PC, Macworld, Mikrodatorn, PC World, Publishing & Design (CAP), DataIngenjoren, Maxi Data, Windows World*; SWITZERLAND's *Computerworld Schweiz, Macworld Schweiz, PC Katalog, PC & Workstation*; TAIWAN's *Computerworld Taiwan, Global Computer Express, PC World Taiwan*; THAILAND's *Thai Computerworld*; TURKEY's *Computerworld Monitor, Macworld Turkiye, PC World Turkiye*; UKRAINE's *Computerworld*; UNITED KINGDOM's *Computing/Computerworld, Connexion/Network World, Lotus Magazine, Macworld, Open Computing/Sunworld*; UNITED STATES' *AmigaWorld, Cable in the Classroom, CD Review, CIO, Computerworld, Desktop Video World, DOS Resource Guide, Electronic Entertainment Magazine, Federal Computer Week, Federal Integrator, GamePro, IDG Books, Infoworld, Infoworld Direct, Laser Event, Macworld, Multimedia World, Network World, NeXTWORLD, PC Letter, PC World, PlayRight, PowerPC World, Publish, SunWorld, SWATPro, Video Event*; VENEZUELA's *Computerworld Venezuela, MicroComputerworld Venezuela*; VIETNAM's *PC World Vietnam.*

HOW TO USE THIS CARD

OUR READER SERVICE CARD HELPS YOU acquire more information about software and hardware products advertised in *DOS Resource Guide.*

Subject to the request of the advertiser, each advertisement that appears in *DOS Resource Guide* has a Reader Service Number (RS#) associated with it. To receive more information about a product:

1. **Note the number that appears in the advertisement.**
2. **Circle the corresponding number on our postpaid card, opposite.**
3. **Be sure to PRINT your name & address on the card, and mail the card today.**

Advertiser	RS#	Page
ACMA Computers, Inc.	*	79
AmRam	*	84
ASCI Computer	134	84
AT&T Multimedia Software	103	7
Buerg Software	109	85
Computer Business Services	146	86
Data Mate	126	86
DataSafe Computing	130	84
Delrina (Canada) Corporation	160	Cov 3
EliaShim Microcomputers, Inc.	153	82
FastTrax International	127	85
Firefly Software Systems Corp.	145	84
FPLOT Corporation	143	84
Harmony Computers	171	80
IBM Corporation	*	Cov 2
Insight Development Corp.	147	84
Mastersoft, Inc.	121	85
MegaHaus Corporation.	150	81
Microsystems Software, Inc.	119	85
Motherboards International	155	83
PowerBASIC, Inc.	111	11
Quarterdeck Office Systems	107	Cov 4
Queblo	136	86
SET, Inc.	131	84
TeraTech, Inc.	114	86
The Computer Book Club	163	87

**This advertiser prefers to be contacted directly.*

FREE PRODUCT INFO!

Want to Know More about Products or Services Advertised In This Issue?

Here's How:

Print your name and address where indicated.

Tell us about yourself by answering the questions.

Circle the numbers on the card corresponding to the reader service number for products and services that interest you.

Tear out and mail the card.

SUBSCRIBE TO

DOS Resource Guide!

Simply check off the box on the reader service card. We'll bill you $23.70* for one year (6 issues).

NO OBLIGATION.

Literature on products and services will be sent to you directly from advertisers,

FREE OF CHARGE.

This rate is limited to the U.S. and its possessions.

DOS RESOURCE GUIDE

READER SERVICE CARD

September 1994
Expires October 31,1994

NAME/TITLE ______________________

COMPANY ______________________

ADDRESS ______________________

CITY ______________ STATE ___ ZIP ______

TELEPHONE ______________________

FAX NUMBER ______________________

100	113	126	139	152	165
101	114	127	140	153	166
102	115	128	141	154	167
103	116	129	142	155	168
104	117	130	143	156	169
105	118	131	144	157	170
106	119	132	145	158	171
107	120	133	146	159	172
108	121	134	147	160	173
109	122	135	148	161	174
110	123	136	149	162	175
111	124	137	150	163	176
112	125	138	151	164	177

☐ **Check here to start your subscription to DOS Resource Guide. You will be billed $23.70* for a one-year subscription (6 issues).**

Is this your copy of DOS Resource Guide? **If YES, Circle #100** **If NO, Circle #101**

B17RS1

DOS RESOURCE GUIDE

READER SERVICE CARD

September 1994
Expires October 31,1994

NAME/TITLE ______________________

COMPANY ______________________

ADDRESS ______________________

CITY ______________ STATE ___ ZIP ______

TELEPHONE ______________________

FAX NUMBER ______________________

100	113	126	139	152	165
101	114	127	140	153	166
102	115	128	141	154	167
103	116	129	142	155	168
104	117	130	143	156	169
105	118	131	144	157	170
106	119	132	145	158	171
107	120	133	146	159	172
108	121	134	147	160	173
109	122	135	148	161	174
110	123	136	149	162	175
111	124	137	150	163	176
112	125	138	151	164	177

☐ Check here to start your subscription to DOS Resource Guide. You will be billed $23.70* for a one-year subscription (6 issues).

Is this your copy of DOS Resource Guide? If YES, Circle #100 If NO, Circle #101

B17RS2

✍ YOU BE THE EDITOR

We hope all of our articles are interesting to read and easy to understand, but help us out by rating the usefulness of some of our columns, departments, and features. In each of the three groups below rank the items, 1 through 4, in order of their usefulness to you. (Number 1 is the most useful to you, number 2 is the second most useful, etc.)

Departments:
- __ Reader's Queue
- __ Q&A
- __ Tips from Readers
- __ Best of the Batch

Columns:
- __ Shareware Exchange
- __ Gookin on DOS
- __ Start-up Clinic
- __ Reader Forum

Features:
- __ Articles about using DOS Commands.
- __ Articles about creating batch files and using batch commands.
- __ General computing (both hardware and software) features (FYI).
- __ Articles about creating and using QBasic programs.

Of all the items mentioned above, which is the most useful: ______________________

____

DRG17

NO POSTAGE
NECESSARY
IF MAILED
IN THE UNITED
STATES

BUSINESS REPLY MAIL
FIRST-CLASS MAIL PERMIT NO. 1007 PITTSFIELD, MA

POSTAGE WILL BE PAID BY ADDRESSEE

DOS Resource Guide
Attn: Reader Service Department
PO Box 5384
Pittsfield MA 01203-9733

NO POSTAGE
NECESSARY
IF MAILED
IN THE UNITED
STATES

BUSINESS REPLY MAIL
FIRST-CLASS MAIL PERMIT NO. 1007 PITTSFIELD, MA

POSTAGE WILL BE PAID BY ADDRESSEE

DOS Resource Guide
Attn: Reader Service Department
PO Box 5384
Pittsfield MA 01203-9733

PLACE
STAMP
HERE

DOS Resource Guide
80 Elm Street
Peterborough NH 03458

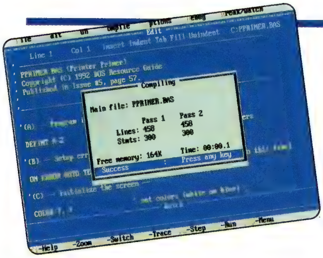
Subscribe Today! 800-349-7327 *Save up to 25% Off* the Cover Price

1-800-349-7327

DOS RESOURCE GUIDE

invites you to

Buy A Great BASIC COMPILER For Only $39.95!



```
' PPRIMER.BAS (Printer Primer)
' Copyright (C) 1992 DOS Resource Guide
' Published in Issue #5, page 57.
```

```
                 Compiling
Main file: PPRIMER.BAS

              Pass 1     Pass 2
                         458
     Lines:   458        300
     Stmts:   300

                         Time: 00:00.1
Free memory: 164K        Press any key
 Success
```

POWER BASIC 2.1

For anyone who has ever written or typed in a BASIC program, this is a remarkable opportunity. Here's a chance to enrich your BASIC programming skills and acquire a powerful BASIC Compiler at a fraction of the regular cost.

HERE'S WHAT YOU GET:

PowerBASIC version 2.1—a complete environment for writing, running, debugging, and compiling BASIC programs that includes:

COMPILER

- MAKES YOUR BASIC PROGRAMS *SUPER FAST!*
- LETS YOU WRITE STAND-ALONE PROGRAMS THAT DON'T NEED QBASIC!
- A COMPLETE, FULL-FEATURED PROGRAMMING SYSTEM

POWERFUL LANGUAGE

Faster and with more features than any other BASIC.

INTEGRATED DEVELOPMENT ENVIRONMENT

Lets you write, edit, run, and debug programs quickly and easily.

GREAT DEBUGGING TOOLS

Helps you find and correct elusive program bugs in seconds.

HELP SYSTEM

Extensive context-sensitive help always available to make learning to program easy and fun.

COMPATIBILITY

Most programs written in other BASICs require only minor modifications.

FREE PROGRAMS

12 BASIC programs from *DOS Resource Guide* to get you started.

YOU WILL ALSO RECEIVE:

"LEARNING BASIC"

A step-by-step tutorial on the ins and outs of PowerBASIC. Written by an experienced BASIC instructor, this book will help you:

- **BECOME FAMILIAR** with the PowerBASIC programming system.
- **LEARN THE BASIC** language one step at a time.
- **WRITE AND EDIT** your own programs, functions and subroutines.
- **DISCOVER TECHNIQUES** for effective troubleshooting.
- **GO BEYOND** the basics, advance your skills.

....AND MUCH MORE!

DON'T MISS THIS UNIQUE OPPORTUNITY! CALL AND ORDER YOUR COPY TODAY:

1-800-349-7327

PLEASE SEND ME: POWER BASIC 2.1

No. of Orders (   ) at $39.95* each

☐ Payment enclosed

Name ______________________

Charge my: ☐ VISA ☐ MC ☐ AmEx ☐ Discover

Address ______________________

City ______________________

Card# ______________________

State ____________ Zip ____________

Expiration Date ______________________

Signature ______________________

Please allow 4-6 weeks delivery. For faster service, please inquire. Prepayment is required on all orders. U.S. funds on U.S. banks only. *Canada, Mexico, and foreign surface, please add $5.00 per item ordered. Airmail, please add $9.50.

Mail To: DOS Resource Guide P.O. Box 672, Mt. Morris, IL 61054

PDRG/001/A17PB

WINDOWS SEAT

A Tale of Two DOS Prompts

by Eric Maloney

As a DOS user, you probably keep an MS-DOS session active during your Windows sessions so that you can run DOS programs. But if you're using Windows' MS-DOS Prompt program information file (PIF) to get to the DOS prompt and run DOS commands, you're not using memory as efficiently as you could. DOS Prompt's PIF allocates far more memory than most DOS commands need, taking resources away from your other Windows applications.

The solution here is to create two DOS Prompt icons, each with its own PIF. The first icon controls a DOS session with just enough memory to run DOS commands; the second manages a session with adequate memory to run larger programs. You keep your "mini-DOS" session active for quick access and use your "maxi-DOS" session only when you need it.

> *MS-DOS Prompt's PIF allocates far more memory than most DOS commands need, taking resources away from your other Windows applications.*

Setting up two icons with two PIFs is simple. First, you keep the standard MS-DOS Prompt PIF, DOSPRMPT.PIF, the same; it will run your maxi-DOS session. Then you create a second PIF called ALTPRMPT.PIF, which uses less memory. Finally, you create an icon for ALTPRMPT.PIF.

Checking Out DOSPRMPT.PIF

Before you create your new ALTPRMPT.PIF, you might want to take a look at DOSPRMPT.PIF. Go to your main program group and click on PIF Editor. Choose File from the menu bar, click on the Open option, and click on DOSPRMPT.PIF.

Note that the Program Filename is COMMAND.COM. This is the DOS program that opens your DOS session; it's the same program DOS runs when you turn on your machine. The Window Title is MS-DOS Prompt. This is the title that will appear under the icon and in the title bar when you're running DOS in a window.

The most important information on screen is in the KB Required and KB Desired boxes that follow Memory Requirements. The standard defaults are 128K and 640K, respectively. (Yours may have already been configured differently, though.) In other words, you can run programs up to 640K in size if the memory is available. Your goal is to create a PIF for an alternate prompt that specifies only 128K for KB Desired.

Creating ALTPRMPT.PIF

Now's the time to create your alternate prompt. While still in the PIF Editor, choose File from the menu bar and click on the New option.

ATTENTION

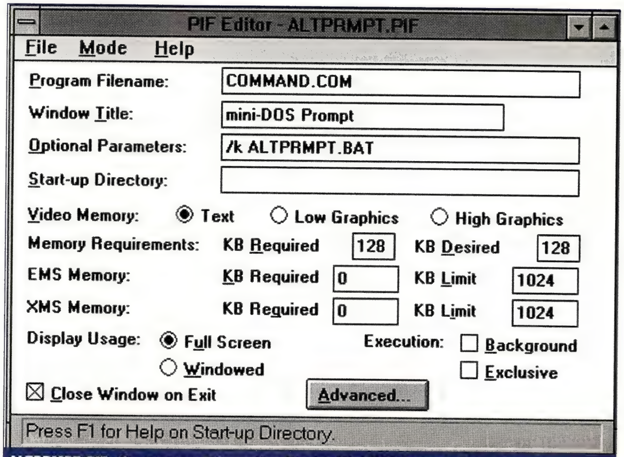
Windows Users

If you use Windows regularly, you probably know a few tricks yourself. Or perhaps you've found a better way to do the job described here. This column is your chance to share your techniques with other Windows users. Contact the Windows Editor at DOS Resource Guide, *80 Elm St., P.O. Box 802, Peterborough, NH 03458-0802, or on CompuServe (75300,2357) or MCI Mail (667-4854).*

- Enter `COMMAND.COM` in the Program Filename box.
- Enter a title in the Window Title box: `MINI-DOS PROMPT`, for example.
- Change KB Desired from 640K to 128K.
- Finally, save the file under the filename ALTPRMPT.PIF.

Your ALTPRMPT.PIF should look like the one in the accompanying screen, except for the information in the Optional Parameters box, which we'll examine here in a moment.

The most important difference between this PIF and the one for your regular DOS Prompt program is the KB Desired setting. Here, 128K gives you enough space to run most DOS commands without hogging memory.



Program Filename: COMMAND.COM
Window Title: mini-DOS Prompt
Optional Parameters: /k ALTPRMPT.BAT
Start-up Directory:
Video Memory: ● Text ○ Low Graphics ○ High Graphics
Memory Requirements: KB Required 128 KB Desired 128
EMS Memory: KB Required 0 KB Limit 1024
XMS Memory: KB Required 0 KB Limit 1024
Display Usage: ● Full Screen ○ Windowed
Execution: ☐ Background ☐ Exclusive
☒ Close Window on Exit
Press F1 for Help on Start-up Directory.

ALTPRMPT.PIF, the program information file for Mini-DOS Prompt. The optional parameter, /K ALTPRMPT.BAT, loads the batch file ALTPRMPT.BAT at the begining of each session; ALTPRMPT.BAT gives you a custom prompt so that you know you're running a mini-DOS session.

A New Icon

Your next step is to create an icon for your alternate prompt. Return to the Windows Program Manager and open the program group that includes your MS-DOS Prompt icon.

Now choose the File option from the menu bar, followed by New. Then select Program Item. A Program Item Properties box will open.

You don't need to enter a description of your alternate prompt, since you already entered a title in the PIF. (If you do enter some other description of the prompt, it will override the PIF title.) Now enter `ALTPRMPT.PIF` in the Command Line box. Click on the OK button. The new icon will appear as part of your program group.

Your alternate prompt works just like your other prompt, except that you're limited to running programs that use less than 128K. That's enough to run the most common DOS commands, including DIR, FORMAT, COPY, and RENAME.

The most notable exception is HELP, which needs almost 300K to manage its data file. If you try to run HELP in your mini-DOS session, you'll get an "Out of memory" message.

A Sense of Where You Are

There's a simple way to include a label in your DOS prompt that lets you know whether you're running a 128K or a 640K DOS session.

For example, when I open a DOS session by clicking on my Mini-DOS Prompt icon, my DOS prompt looks like this:

```
[MINI-DOS] C:\>
```

First, create a batch file called ALTPRMPT.BAT that includes the following line:

```
PROMPT [MINI-DOS] $P$G
```

Insert your own descriptive label in place of [MINI-DOS]. Save the batch file in your Windows subdirectory. Next, using the PIF Editor again, open ALTPRMPT.PIF. Include the following line in the Optional Parameters box:

```
/K ALTPRMPT.BAT
```

/K is a standard DOS COMMAND.COM switch; it tells COMMAND.COM to run the program or batch file that immediately follows. When Windows runs COMMAND.COM to open the DOS session, it will pass the switch along to COMMAND.COM, which will then run your ALTPRMPT.BAT file.

You might want to use the same technique to create a similar label for your standard DOS Prompt sessions. ■

Eric Maloney is the former editor in chief of 80 Micro, *the founding editor of* PC Resource, *and the former editor in chief of* LAN Technology *magazine. He is* DOS Resource Guide*'s editor at large.*

(Continued from page 88)

In addition, when you register BATMNU, you can order a Menu Utility program that lets you create menus almost painlessly. You can also order from the author a CD-ROM containing 2,300 shareware programs for $9.95, half the disc's normal price.

Although it doesn't include all the fancy features of the following two programs, BATMNU is easy to use, runs quickly, and seems to do its job well. My only real complaint is that BATMNU doesn't support 43- or 50-line mode on EGA and VGA systems.

A Good Mouser

The next batch-file utility comes from Bruce Coller of Ashburnham, Massachusetts:

> *I've been trying ways to use mouse input in batch files. I'd like to recommend **BTX 1.0** as a solution to this problem. BTX includes both mouse and keyboard input along with a variety of colors and sounds. Each menu can be defined on a single batch-file line that can't exceed the DOS limit of 127 characters per line. You can overcome this limitation, however, with 4DOS (from J.P. Software) or NDOS (in Symantec's Norton Utilities).*

BTX is unusual in that you control it by choosing from some 40 switches, or command-line options. Although BTX can display a list of these options, you'll need to peruse the manual to understand all of them.

Luckily, you don't need switches for many menus. If you want to create a menu with the choices *First*, *Second*, and *Third*, you can use a line like this in your batch file:

```
BTX FirstF SecondS ThirdT
```

The *F*, *S*, and *T* at the end of the words in that line are the "hot keys" the user can press instead of clicking a mouse or moving a cursor bar. That's pretty simple, and it makes for an attractive menu.

The various switches let you run BTX in screen-saver mode; display help text; set borders, background characters and colors, margins, and shadows; select defaults; and add headers, footers, titles, and help information, among other things.

For complex menus, you won't have enough room on the command line for everything, even if you're using 4DOS or NDOS with their double-length command lines. Instead, you can tell BTX to read menu and help information from data files. This isn't a disadvantage, since most menu utilities use menu files exclusively.

Once you register BTX with its developer, you can have it display your name and your company's name instead of the developer's name. That makes the program ideal for consultants and developers who might want to install BTX on a client's machine or use it on a laptop.

Also, BTX is smaller than many batch-file utilities. After you configure it with your name, you can use a utility such as LZEXE (a shareware file-compression program) to squeeze it down to under 25K bytes. That means it can fit comfortably on even a 360K floppy disk.

Like BATMNU, BTX doesn't support 43- or 50-line mode on EGA and VGA systems. Am I the only one who runs DOS in this mode most of the time?

PRODUCT INFORMATION

Prices shown are for registration only. Most vendors charge additional small fees for shipping and handling, extra disk copies, and printed documentation.

A Batch View 1.1, $15
Robin R. Latham
P.O. Box 16152
Fort Worth, TX 76162
817-263-4056

BATMNU 2.3, $15
Warren Small
35 Benjamin St.
Manchester, NH 03109
(mail inquiries only)

BTX 1.0, $12
E. Kasey Kasemodel
401 Snyder Ave.
Ann Arbor, MI 48103
(mail inquiries only)

Are You
WIRED?

If you're wired (or only a little uptight) and you do some modem communicating, stop into the ***DOS Resource Guide*** Bulletin Board. You'll find all of our QBasic programs available for downloading, as well as all of the shareware products mentioned in our pages.

The DRG Bulletin Board is also the fastest way to get in touch with the editors.

Let us hear from you:
(603) 924-3181

Seven Utilities in One

Our third and final batch utility, at least for this *DRG* issue, is **A Batch View**, nominated by Craig Ewasko of Edmonton, Alberta, Canada:

> *A Batch View version 1.1 is an excellent utility for batch files. It includes menu making and mouse support, a scrolling screen clearer, and more. You can create professional-looking batch files in seconds, it's easy to use, and it costs only $15 to register.*

A Batch View is actually eight executable (EXE) programs that work together to enhance batch files. You can use them individually or together to create unusual and powerful batch files.

The first program, ABATMENU.EXE, creates pop-up menus that are centered automatically on screen and show 3-D shadows. Each can include up to 48 menu items the user can select with the cursor bar or a mouse. This program doesn't support hot keys, however—a serious shortcoming, in my opinion, especially if the user brings up the same menus daily and knows in advance what they're going to say.

AMSG.EXE lets you create user prompts that accept either a yes/no response or up to 64 characters of input. In the latter mode, AMSG stores the input in the environment so that your batch file can examine the response. The same program can display the time, disk status, and a complete list of environment variables.

AECHO.EXE is an enhancement of two DOS commands: ECHO and MORE. It lets you position text on screen, control the color of the text, and left- or right-justify a block of text.

AMSCROLL scrolls text around, beneath, above, or through the screen in a variety of ways until the user presses a key. It's a glitzy form of DOS's PAUSE command. ABATINFO collects information about DOS, the date, your disk drives, and your files and returns that information to a batch file.

ABATJOB and ACLS both control text on screen. ACLS, an enhancement of the DOS CLS command, is particularly interesting because it features ten different modes of clearing the screen and can also set screen colors. Finally, APAGE lets the user view ASCII files and scroll through them.

It's a long list of utilities, most of which work well. I found problems, however, with those that access the DOS environment, persisting whether or not Microsoft Windows was running and whether or not I used J.P. Software's 4DOS or DOS's standard COMMAND.COM as my DOS shell. Some were caused by the number and length of entries in the environment.

Other than those bugs, A Batch View worked well. I find the program less satisfying than BTX and BATMNU, in part because it's less focused, in part because of its size (the eight EXE files use more than 160K of disk space), and in part because I already have utilities that perform many of this program's functions. And I guess I'm just not as impressed with the visual effects that A Batch View can create as I used to be.

But don't let me discourage you from trying this or the other batch-file utilities. One of the three may be just right for your needs. ■

Contributing Editor Hardin Brothers evaluates shareware programs for CompuServe's electronic magazine On-Line Today.

SHARE THE WEALTH

"Shareware Exchange" invites you to send a copy of your favorite shareware program, along with a description and an explanation of why you like it (500 words or less), to Shareware Editor, *DOS Resource Guide*, 80 Elm St., P.O. Box 802, Peterborough, NH 03458-0802. Be sure to tell us how we can obtain a copy of the program. We'll pay contributors $50 for each program featured in this column. Please don't send us shareware you've written—we want recommendations from users, not authors. You can contact us on CompuServe at 75300,2361 or on the *DOS Resource Guide* bulletin-board system (BBS) at 603-924-3181.

All programs featured in "Shareware Exchange" are available electronically from the *DRG* BBS. There are no connect-time charges—you pay only for the phone call. Set your communication program's parity to 8 data bits, no parity, 1 stop bit (8,N,1). Then dial 603-924-3181 and wait for the "Connect" message, which tells you that you've entered the BBS successfully. If you're a first-time user, the system will ask you to verify the spelling of your name and to choose a password. Then it will display a general information screen, followed by a questionnaire containing ten questions about your address, phone number, and so on. (You must answer the questionnaire to set up your user account and gain access to the BBS.)

From this point on, the on-screen prompts are the same for all users. Press C (for Continue) at the next screen to access the main menu. From there, press the F key (for the File menu) and Enter, followed by L (for List Available Files) and Enter.

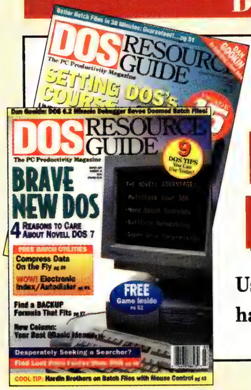
Then you'll see a list of file areas. Choose the number (2 through 5) corresponding to the shareware category listed for each program in this article, and follow the downloading instructions.

A Batch View 1.1, BATMNU 2.3, and BTX 1.0 are all in Shareware Utilities. In addition, most of the programs described in "Shareware Exchange" can be found on major on-line services and local BBSs.

—Eds.

THE COMPLETE DOS RESOURCE GUIDE

BACK ISSUES ORDER FORM



Make sure you own all the best

BACK ISSUES!

Use our toll-free number and have your credit card ready!

DRG#5: The One-Stop DOS Primer
All you can eat: DOSKEY, SETVER, APPEND, and ASSIGN, plus a tutorial on using the ATTRIB command. Part 1 of a DEBUG series. Our "10 Worst" list of DOS commands. Your introduction to serial ports. FREE QBasic printer control program: just type and run. Sharpen your DOS finesse with 15 type-in tips.

DRG#6: 13 Ways to Boost Your DOS IQ
Peter Norton shares some valuable COPY secrets. Personalizing the DOS 5 shell. Part II: altering disk data with DEBUG. Modem communications made simple. Meet the DOS competition—our first look at DR DOS. Compression/decompression software explained. Part I of a batch-file clinic. FREE PROGRAM: a QBasic name and address database. Plus, 13 tips for boosting your DOS IQ.

1-800-349-7327

PDRG/001/A17BA

BACK ISSUES of DOS Resource Guide are only $5.95* postpaid!

Please enter below the quantity of each Back Issue needed.

DRG#5 D005		**DRG#9** D009		**DRG#13** D013	
DRG#6 D006		**DRG#10** D010		**DRG#14** D014	
DRG#7 D007		**DRG#11** D011		**DRG#15** D015	SOLD OUT
DRG#8 D008		**DRG#12** D012		**DRG#16** D016	

I enclose $5.95 for each issue. Please mail (post paid) to:

Name ____________________

Street ____________________

City ____________________

State/Zip ____________________

☐ Payment enclosed ☐ Check or MO

Charge my: ☐ VISA ☐ MasterCard ☐ AmEx ☐ Discover

Card # ____________________

Exp. Date ____________________

Signature ____________________

*Back issues to Canada and Mexico cost $6.95 (postpaid); other international (surface) $8.00, (air) $13.00. Checks and money orders must be payable in U.S. funds, drawn on a U.S. bank.VISA, MasterCard, Discover, and American Express also accepted. Please include your card number, its expiration date, and your signature on your request. Please indicate which issue(s) of DOS Resource Guide you are ordering.

DRG#7: Get Creative with DOS!
Instant programming with articles on DEBUG and batch files: You'll learn how to write your own DEBUG scripts and add error-code checking to a batch file. How to outline a QBasic program. Should you divvy up your new hard drive? Our picks for building a shareware library and advice on how to avoid 35 traps in a Windows setup. Plus, 15 tips to DOS happiness.

DRG#8: Getting CONFIG.SYS Right!
John Dvorak takes you step-by-step through the setup of CONFIG.SYS. Put personality in your PROMPT. A MIRROR command can save your data. Taking care of the environment in a batch file. FREE PROGRAM: A perpetual calendar in QBasic. How to customize your modem. More on PATH, pens, and Windows Program Manager. Plus, our annual index to articles: DRG 1–7. And 8 tips add power to your command line.

DRG#9: Create Your Own DOS Game!
3 FREE games in QBasic: type and run. First of a series on ways to improve batch files using DEBUG. Create a database with the prompt command. Switch-hitting with DOS 5's DIR. Dan Gookin's new FORMAT.COM. 9 batch files from Ronny Richardson's collection that will move you from one subdirectory to another. 7 shareware utilities that make life easier at the DOS prompt. Plus 6 tips from the command line.

DRG#10: Are You Ready for DOS 6?
First look at MS-DOS 6: Double the disk storage, better backup. All about DELETE. Automate CHKDSK in a batch file. Memory management special: how DOS gets out of 640K, how memory utilities work faster and easier. Peter Norton gets your DOS applications started under Windows. FREE PROGRAM: Fraction Maker — decimal to fraction/fraction to decimal converter. How to beat hard-drive fatigue. 7 tips from the prompt.

DRG#11: Are DOS Utilities Doomed?
Utility publishers struggle successfully to outperform DOS 6 with its caching, memory, and hard disk add-ins. FREE: DEBUG script keeps checking for input. An easy batch file for killing a directory without DOS 6. Converting GW-Basic programs to QBasic. Q&A on Super VGA. Using the new CHOICE command. FREE: QBasic loan calculator. 8 DOS tips for the command-line insecure.

DRG#12: Setting DOS's Course
The DOS environment statement is easier to understand than you think. All about FORMAT. A DEBUG script that launches a program after a countdown. Get started with batch files: Guaranteed! Disk caching explained. Using ERRORLEVELs in batch files. Gookin on DOS 6 DEFRAG. FREE: QBasic color finder. Plus 15 power tips from the prompt.

DRG#13: The DOS Wars Heat Up
IBM PC DOS vs MS-DOS. 10 ways to use DOSKEY. DoubleSpace changes how you use other commands. How to make batch files a little friendlier. Where to find DOS help on-line. Advanced class in QBasic—boost program speed with assembly-language assists. Index to DRG's DOS tips: Issues 1–12. Plus 13 new tips for the self-taught DOS pro.

DRG#14: Brave New DOS
4 reasons to care about Novell DOS 7. 2 FREE batch utilities: electronic Rolodex/autodialer and do-it-yourself data compression. Using MS-BACKUP. Redirecting input with a DEBUG script. Adding mouse control to a batch file. Debug batch files line by line. FREE PROGRAM: Hangman! Plus, 9 DOS tips for the C-prompt aficionado.

DRG#15: SOLD OUT!

DRG#16: Super DOS Utilities
2nd annual guide to DOS utilities—analyzes over 300 specialized products for DOS users. Dan Gookin on Windows and DOS alliances, a lesson in using FOR-IN-DO in a batch file. Reviving your drive with SCANDISK. Using UNDELETE in MS-DOS 6.x. FREE PROGRAM: QBasic menu creator. Multi-boot batch file for pre-MS-DOS 6 owners. Swap keys in any DOS program with DEBUG. 9 tips from the command line.

TOLL-FREE 1-800-349-7327 TO ORDER!

Send your check or money order to: DOS Resource Guide P.O. Box 672 Mount Morris, IL 61054

PC WORLD
BEST BUY ~~FIVE~~ 6 TIMES

The Best Buy for Custom Built Desktop System $1995.

PC WORLD BEST BUY
May 1994
June 1994

PC WORLD BEST BUY
June 1993
February 1993
October 1992
November 1992

MAY & JUNE

PC WORLD BEST BUY
PC WORLD
BUDGET DESKTOPS

486DX2 66MHz with PCMCIA Card Drive

8MB RAM
VESA Local Bus
Super VGA Card with 1MB
Mini Tower Case
UL/CSA Power Supply
2 Floppy Drives
PCMCIA DUAL CARD DRIVE
340MB Hard Drive
15" SVGA Digital Controlled Monitor
Microsoft Windows & DOS

Call today for your $1995 Best Buy

ACMA 486 SERIES

Call for detailed configuration.

SX33 $1195	**DX2 50** $1425	**DX4** CALL
DX33 $1395	**DX2 66** $1495	

energy
EPA POLLUTION PREVENTER

Ask about Green Computers

ACMA Multimedia SERIES

Dual Speed CD-ROM
16 Bit Sound Card
Built-in 15 Watt Speaker
15 Watt Amplifier
CD-ROM Software Titles
*Call for detailed configuration.

SX33 $1495	**DX2 50** $1725
DX33 $1695	**DX2 66** $1795

ACMA Pentium SERIES

PCI Local Bus
32 Bit Processor
PCI Local Bus Video Card
*Call for detailed configuration

60MHz	**66MHz**
$2195	$2295

Onsite Service - 45 Money-Back Guarantee - Advance Replacement Program

800-786-6888
Ext. 888

Acma COMPUTERS, INC.
47988 Fremont Blvd., Fremont, CA 94538
(510) 623-1212 FAX (510) 623-0818

Prices and specifications subject to change without notice.
Prices do not include shipping & handling. Call to confirm configuration.
Intel Inside logo is a registered trademark of Intel Corporation.

14 YEARS OF GREAT PRODUCTS, GREAT SERVICE & GREAT VALUES!

LASER PRINTERS

Panasonic 4410/4430	489/639
Panasonic 5400/4400	*699/429
Panasonic 4440	959
Canon LBP 430	*529
Canon LBP 860	*1199
H.P. 2P Plus (with toner)	599
H.P. 3P (with toner)	799
H.P. LaserJet 4L (with toner)	649
H. P. LaserJet 4P (with toner)	959
H.P. LaserJet 4 (with toner)	1279
H.P. LaserJet 4Plus(with toner)	1399
H.P. LaserJet 4MPlus (with toner)	1929
Okidata 400 E/410E	494/649
Okidata OL 810 LED	899

PRINTERS

Canon BJ-10SX	*199
Canon BJ-100	*219
Canon BJ200E	*249
Canon BJ230	*359
Canon BJC 600 Color	*519
Canon BJ-Color 820	*999
Citizen PN48 Portable (Pro)	189
Citizen Notebook Printer II color	295
Citizen ProJet II	*239
Citizen GSX 220 w/color kit	179
Diconix 180SI Ink Jet Portable	214
Epson Stylus 300/800	209/265
Epson LQ570+/LQ1070+	239/379
H.P. Desk Jet 310	295
Hewlett Packard Desk Jet 520	289
Hewlett Packard Desk Jet 550 Color	419
Hewlett Packard Desk Jet 560	559
Hewlett Packard Desk Jet 1200 Color	1389
Okidata 184 Turbo/380	219/215
Okidata 320/321	295/419
Okidata 520/521	365/484
Okidata 590/591	419/574
Panasonic 1150/2023	126/169
Panasonic 1123/1695	179/389
Panasonic 2123/2124	219/289
Panasonic 3123/3124	249/319
Panasonic 2130/2135	199/239
Panasonic 1624/2624	359/379
Seiko Smar t Label Plus/Pro	139/229
Star 245OR/248OR	229/259
Star 1001/1040R	134/159
Star SJ 144 Color	449

*after rebate

SOFTWARE

Aldus Pagemaker 5.0 Windows	489
Omnipage Direct/Pro 5.0	99/439
Lotus for Windows Rel. 4	249
Lotus 123 Version 3.4A/2.4	339/289
Microsoft Word for Windows	279
Microsoft Windows 3.0/3.11	29/95
Norton Utilities 8.0	99
Word Perfect 6.0 For DOS	249
Word Perfect (6.0A) Windows	239
PC Tools for Windows	89
Q&A 4.0 DOS/Windows	189/149

MONITORS

NEC MultiSync 3V/4FGE	459/599
NEC Multisync 5FGE	969
NEC Multisync 5FGP	1079
Panasync 1395 SVGA NI	279
MAG LX1564	359
MAG MX15F	459
MAG MX 17FG	829
MAG DX15F/DX17F	419/689
MAG 17FG	919
Sony GDM 17 SEI	1099
Viewsonic 15EX	369
Viewsonic 17	929
Viewsonic17G/17E	839/769
Harmony 17" (1280 x1024)	589
Harmony SVGA .28 Non-Interlaced	245
Harmony SVGA .28(1024x768)	225
Harmony VGA .39 (1024 x 768)	199
Harmony Mono VGA/TTL	119/119

NOTEBOOK COMPUTERS

Toshiba 4700CS DX-2 50Mhz 200 Meg/320 Meg	3139/3389
Toshiba 1950 486 DX-2 40Mhz	call
Toshiba 1950 CS 200 Meg	call
Toshiba 1950 CT Color 200 Meg/320 Meg	2899/3149
Toshiba 1910 486SX33, 4 Meg Ram, 200 Meg	call
Toshiba 1910 CS Color 120 Meg/200 Meg	1949/2039
Hyundai 486SL 25 2 Meg RAM, 120 Meg	1099
Canon NoteJet 486SX-33, 4Meg Mono 120/200	1799/1899
Color 12/200	2389/2479
NoteJet II 486SL C2/50 Mhz 4Meg Color130/260	2689/2889
IBM Thinkpad 350/360	call

HARD DRIVES & TAPE BACKUPS

Seagate 107 Meg/129 Meg	179/189
Seagate 214 Meg/260 Meg	225/239
Seagate 341 Meg/457 Meg	285/339
Colorado Trakker 250	289
Colorado Tape Backup 250	154
Colorado Trakker 120/250	199/289

VIDEO CARDS

ATI Graphics Pro Turbo 2Meg	369
ATI Graphics Pro Turbo 4Meg	549
ATI Graphics Ultra Pro 2Meg	249
Ultra Pro 2Meg No Mouse	239
ATI Ultra Pro 2 Meg (PCI)	279
ATI Ultra Pro 2 Meg (EISA)	329
ATI Graphics Ultra Plus 2Meg	189
Ultra Plus No Mouse 2Meg	179
ATI Graphics Ultra 1Meg	129
ATI 8514 Ultra 1Meg	139
ATI Graphics Xpression 2Meg	229
ATI Graphics Wonder 1Meg	144
Hercules Power Graphite 1 M/2M	224/339
Hercules Dynamite Pro 2 Meg	189
Dynamite (PCI) 1 Meg/2 Meg	194/249
Graphite Pro 1 Meg/2 Meg	234/359
Diamond Stealth Pro 1 Meg	219
Diamond Stealth Pro 2 Meg	289
Diamond Speed Star 64 2M/4M	169/209
Diamond Viper 2 Meg (PCI)	349
Diamond Speed Star Stealth 24	139
Harmony VGA (1024 x 768) 1 Meg	69
Harmony VGA Accelerator	79

VESA (Local Bus)

ATI Graphics Wonder	144
ATI Xpression 2 Meg	229
ATI Graphic Ultra Plus 2 Meg	189
ATI Graphic Ultra Pro 2 Meg	249
ATI Graphic Pro Turbo 2 M/4M	369/549
Diamond SpeedStar Pro (1Meg)	119
Diamond Stealth 24	139
Diamond Stealth 32 2 Meg	219
Stealth Pro 1Meg/2Meg	214/284
Diamond Viper (2 Meg)	324
Viper Pro 2Meg/4Meg	379/559
Diamond Stealth 64 2Meg/4Meg	319/459
Hercules Dynamite Pro (1Meg)	155
Hercules Dynamite Pro (2Meg)	189
Hercules Dynamite Power 2 Meg	224
Hercules Stingray (1Meg)	129
Hercules Power Graphite 1M/2M	224/339
Harmony VGA Accelerator (1Meg)	99

MULTIMEDIA

Mitsumi CD-Rom Internal 350 m.s.	129
Mitsumi CD-Rom Double Speed	169
NEC CD-Rom 3XI/3XE	444/479
Media Vision Fusion CD LX	269
Creative Labs OMNI CD	209
Creative Labs SB16 Discovery Kit	309
SoundBlaster 16 MCDASP	169
Soundblaster 16ASP (SCSI)	149
Soundblaster 32 Adv. Wave Effect	279
Soundblaster 16 SCSI	199
Sound blaster 32 AWE (ASP)	279

MODEMS & FAXBOARDS

Hayes Optima 14.4 Ext. FaxModem	339
Hayes Optima 28.8 Ext FaxModem	409
Hayes Accura 14.4FaxModem internal/external	149/159
Hayes Accura 28.8 Ext.	259
ATI 2400 ETC. Internal/External	59/79
ATI 9600 ETC/E	129
ATI 14.4 FaxModem Int/Ext	165/185
Intel #100/#200	99/209
Intel #400/#400E	259/299
Intel 14.4 Faxmodem Int/Ext	119/139
Zoom 14.4 FaxModem Int/Ext	109/129
Zoom 14.4 Voice Int/Ext	149/169
Zoom 28.8 FaxModem Int/Ext	194/219
Boca 14.4 Faxmodem Int/Ext	94/129
Boca 14.4 Voice Int.	149
Boca 28.8 Faxmodem Int/Ext	189/219
Practical Peripherals 14.4 Faxmodem Int.	114
Practical Peripherals 14.4 Ext. Mini	139
Practical Peripherals 28.8 Int./Ext.	279/299
U.S. Robotics Sportster 14.4 FaxModem Internal/External	109/129
U.S. Robotics 14.4 PCMCIA	199
MegaHertz XJack PCMCIA FaxModem 9624-14.4	139/209
96/24 FaxModem Int./Ext.	49/69
Harmony 28.8 Internal	169
Harmony 14.4 FaxModem Internal/External	89/119
Pocket 14.4 FaxModem	179
Harmony PCMCIA FaxModem 96/24	119
Harmony PCMCIA FaxModem 14.4	189

SCANNERS/ MICE/KEYBOARDS

HP ScanJet IIP/IICX	479/949
Microtek Scanmaker 2G/2SP	459/749
Canon IX-4015 Scanner (after rebate)	849
Microsoft Mouse (serial)	59
Microsoft Ball Point Mouse	79
Logitech Mouseman/Trackman	44/74
Logitech Mouseman Cordless II	84
Logitech Trackman Voyager	69
Logitech Scanman 32/256	89/149
Logitech Scanman Easytouch	189
Logitech Scanman Color	339
Scanman Powerpage	469
Harmony Mouse/Trackball	29/39
Harmony PS/2 MOUSE	39
Keytronic 101 keyboard	49
101 Keyboard w/trackball	39/69
Focus FK2001 Keyboard	49

■ Major credit cards welcome at no additional charge.■ Shipping & handling extra.
■ Defective merchandise will be replaced or repaired at our discretion within the terms of our warranty. ■ Price and availability subject to change without notice. ■ We do not allow non-defective returns. For defective merchandise inquiries call customer service at (718) 692-2828. ■ We are not responsible for typographical errors ■ Authorized Panasonic Service Center.

SALES DEPARTMENT
OPEN SUNDAYS
SUNDAY 10:30-4:30 EST
MONDAY-THURSDAY 9-6 EST
FRIDAY 9-3:30 EST
718-692-3232

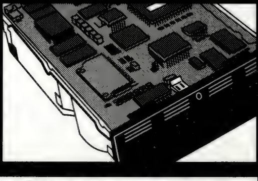
Customer Service 718-692-2828
1801 Flatbush Avenue, (bet. Aves J&K)
Brooklyn, NY 11210

HARMONY COMPUTERS

800-441-1144

Circle 171 on Reader Service Card

MEGAHAUS
HARD DRIVES
The Drive Authority!



UNEQUALED REPUTATION

Nobody knows drives like MegaHaus. We have specialized in storage solutions for over 8 years. Repeat business from satisfied customers is the backbone of our business. We satisfy our customers.

DIRECT REPLACEMENT WARRANTY

During the first year of your warranty, we replace: not repair, any defective items. All items receive lifetime technical support from our factory trained techs.

SAME DAY SHIPPING

Our multi-million dollar inventory offers you the best diversity and availability in the market. We ship most orders the same day. Each order is checked at least four times for accuracy.

FAST, EASY INSTALLATION

Anyone can sell you a hard drive. We sell easy to install storage solutions. We spend countless hours researching the best solutions, then create our quick precise installation guides for each kit.

UNBEATABLE DEALS

WESTERN DIGITAL

#1 IDE drives. 3 year warranty, 128K read / write disk cache. MegaHaus unbeatable price!

420MB	12ms	WDAC2420	$225
540MB	12ms	WDAC2540	$315
730MB	10ms	WDAC2700	$425

MICROPOLIS

Best buy large capacity drives! 5 year advanced replacement warranty. MegaHaus unbeatable price! Choose IDE or SCSI.

1050MB	8.5ms	MC4110	$649
1740MB	10ms	MC2217	$899

Seagate

Baracuda drives, 7200rpm, 1MB buffer. FAST! MegaHaus unbeatable price! WIDE SCSI in stock.

1690MB	8ms	ST11950N	$1249
2139MB	8ms	ST12550N	$1329
4290MB	8ms	ST15150N	$2389

FUJITSU

JAPANESE QUALITY! INCREDIBLE PRICE!

240MB	2.5"	14ms	IDE or SCSI	1yr wty.	$299
530MB	3.5"	13.8ms	IDE	3yr wty.	$325
530MB	3.5"	13.8ms	SCSI	3yr wty.	$335
1080MB	3.5"	9.5ms	SCSI	5yr wty.	$599

IOMEGA

INCLUDES 1 FREE TAPE! FIVE YEAR WARRANTY!

250MB tape backup. $10 drive discount w/ hard drive sale.

Internal (mounts in 3 1/2" or 5 1/4" drive bay)	$159
External for printer port	$319
3 Pack of Preformatted DC2120 tapes	$40

WIDE SCSI

WE'RE THE EXPERTS!

Includes drive (5yr warranty), controller, & cable.

	1Gig	2Gig	4Gig
PCI	$1099	$1729	$2899
VESA or EISA	$1199	$1829	$2999

HARD DRIVES

CONNER

IDE			
210MB	14ms	CFS210A	$182
420MB	14ms	CFS420A	$232
541MB	12ms	CFA540A	$316
811MB	10ms	CFA810A	$445
1080MB	9ms	CFP1080A	$569

Maxtor

IDE			
170MB	15ms	7171A	$172
540MB	12ms	7546A	$285

MICROPOLIS

IDE			
1050MB	10ms	MC4110A	$649
1740MB	10ms	MC2217A	$899
SCSI			
1052MB	8.5ms	MC4110	$649
1740MB	10ms	MC2217	$899
3020MB	11ms	MC1936	$1899
SCSI-AV Audio / Visual tuned drives			
1000MB	10ms	MC2210AV	$749
1700MB	10ms	MC2217AV	$999
3000MB	11ms	MC1936AV	$2119

Seagate

IDE			
213MB	16ms	ST3243A	$179
260MB	16ms	ST3290A	$205
341MB	13ms	ST3391A	$222
420MB	14ms	ST3491A	$235
545MB	12ms	ST5660A	$319
SCSI (WIDE SCSI also available)			
1690MB	8ms	ST11950N	$1249
1700MB	9ms	ST11900N	$1079
2145MB	8ms	ST12550N	$1329
2148MB	9ms	ST12400N	$1189
4290MB	9ms	ST15230N	$2299
4290MB	8ms	ST15150N	$2389

WESTERN DIGITAL

IDE			
341MB	12ms	AC2340	$219
425MB	12ms	AC2420	$225
540MB	11ms	AC2540	$315
730MB	10ms	AC2700	$425
1083MB	10ms	AC31000	$575

HARD DRIVES

Quantum

IDE			
270MB	12ms	LPS270A	$209
342MB	12ms	LPS340A	$233
541MB	12ms	LPS540A	$309
SCSI			
170MB	14ms	ELS170S	$199
270MB	12ms	LPS270S	$225
342MB	12ms	LPS340S	$265
541MB	12ms	LPS540S	$325
1080MB	10ms	IMP1080	$665
1440MB	9.5ms	IMP1440S	$850
2160MB	9.5ms	IMP2160S	$1165

IBM

IDE			
270MB	12ms	70G8487	$209
364MB	12ms	70G8488	$227
527MB	12ms	70G8486	$322
728MB	12ms	70G8511	$399
1052MB	8.6ms	70G8512	$699
SCSI			
270MB	12ms	94G2439	$220
364MB	12ms	94G2440	$239
527MB	12ms	94G2441	$339
728MB	12ms	94G2422	$449
1052MB	8.6ms	70G8492	$679
2040MB	9.2ms	70G8493	$1199
4250MB	11ms	90F0780	$2239

SYQUEST

SCSI		Drive	PC Kit	Mac Kit
44/88MB	SQ5110C	$269	$419	$399
105MB	SQ3105S	$239	$389	$329
200MB	SQ5200C	$399	$609	$519
270MB	SQ3270S	$369	$559	$489
IDE		Drive	PC Kit	Portable
105MB	SQ3105A	$179	$279	$359
270MB	SQ3270A	$369	$495	$599

CARTRIDGES Qty 1 / 5 or with drive

88MB $85/75 105MB $65/60 200MB $99/89 270MB $92/82

MISCELLANEOUS

Macintosh Drives Hard Disk Cards
PCMCIA drives Floppy Drives
IBM PS/1, PS/2, and Tandy Drive upgrades

SCSI CD ROM DRIVES

NEC	CDR900	4X	180ms	$939
Plextor	PX43CH	4X	150ms	$475
NEC	CDR510	3X	195ms	$345
Toshiba	XM3401	2X	200ms	$269
Toshiba	XM4101	2X	320ms	$215
Teac	CD50	2X	265ms	$209
Media Vis.	SH93R	2X	280ms	$159

HOT DEAL, Macintosh external $255

Pioneer External CD Changers (300ms)

DRM602X	6 disc	307KB/sec	$699
DRM604X	6 disc	612KB/sec	$1049
DRM1804X	18 disc	612KB/sec	$2079

MULTIMEDIA KITS

SCSI Drive, sound card (in retail pkg)
PREM=Media Vision Premium 3D. PRO=Media Vision Pro 3D
SB= Sound Blaster 16 SCSI. SA= Sound Blaster SCSI ASP.

	PREM	PRO	SB	SA
SH93R	$309	$445	$309	$379
XM4101	$365	$489	$359	$425
CD50	$359	$485	$359	$425
XM3401	$425	$549	$425	$479
CDR510	$499	$619	$499	$559
PX43CH	$659	$779	$659	$699
CDR900	$1129	$1249	$1129	$1189

CONTROLLER CARDS

ADAPTEC	Alone	w/Drive
A1522BASIC 16Bit SCSI	$119	$109
A1542CBASIC 16Bit SCSI	$225	$215
A2742 EISA SCSI	$319	$309
A2842 VESA SCSI	$279	$269
FUTURE DOMAIN		
1680SVP 16Bit SCSI	$155	$139
QLOGIC PCI SCSI & WIDE	$325	$309
PCI 4IDE, incl. CD ROM & FAST ATA	$59	$49
ALPHA RESEARCH CACHING CARDS		
S425I 16Bit IDE	$99	$89
S475I 32Bit Local Bus IDE	$139	$129
S485I 32Bit PCI IDE	$229	$219
S425S 16Bit SCSI	$229	$219
S475S 32Bit Local Bus SCSI	$369	$359
GSI 16Bit IDE accelerators (enhanced IDE)		
1533-21 2IDE, 4 Floppy	$65	$55
1533-32 2IDE, 4 Floppy, IO	$129	$109
4C 16Bit supports 8 IDE	$139	$119

Many other types of controllers in stock.

SCSI TAPE DRIVES

Kit includes 16bit card, DOS or Windows software & 1 tape.

			Bare	Kit
250MB	Conner	C2150	$349	$569
525MB	Conner	C2525	$489	$699
580MB	Teac	MT01F	$399	$599
750-1.5Gig	Exabyte	2501	$499	$699
2Gig	Conner	4320	$749	$899
2Gig	HP	35470	$889	$999
4-8Gig	Conner	4324	$999	$1109
4-8Gig	HP	35480	$989	$1099
4-16Gig	Sony	SDT5000	$1029	$1169
4-16Gig	Conner	4326	$999	$1139
4-16Gig	HP	C1533A	$1379	$1579

NovaBack backup software for SCSI

Novell NLM 50 user	$229
Novell NLM unlimited user	$389
NovaBack OS/2	$89
NovaBack Macintosh	$109

SMART STORAGE SOLUTIONS

Smart STORAGE SOLUTIONS

Smart Storage Solutions manufactures quality storage upgrades, and offers them at aggresive prices. All products include toll free support!

OPTICAL DRIVES

High quality, high performance optical drive solutions include 1 disk. Ask for your free "Dare to Compare" fax.

	128MB	230MB	1.3Gig
Internal upgrade	$679	$879	$2209
Internal Kit	$769	$969	$2299
External for Mac	$699	$899	$2269
Portable for PC	$899	$1099	$2469

Other configurations available.

CD ROM DRIVE TOWERS

Multiple CD ROM drives in a single external enclosure. Great for networks, BBS, or business.

	2 Drive	4 Drive	7 Drive
Media Vis. 93R	$429	$939	$1539
Toshiba 4101	$569	$1199	$1999
Teac CD50	$569	$1199	$1999
Toshiba 3401	$669	$1499	$2499
NEC 3X	$859	$1829	$3099
Plexstor 4X	$1089	$2529	$4299

Other configurations available.

MEGAHAUS
HARD DRIVES
1110 NASA Rd 1 #306
Houston, Texas 77058

1-800-786-1183

Ad. Code # DW10 Sales Hours (Central Standard Time)
8am - 7pm Monday thru Friday 9am - 3pm Saturday
FAX (713) 333-3024 Local (713) 333-1910
We sell within the United States, its territories, and Canada (sorry no international orders)

VISA MasterCard DISCOVER AMERICAN EXPRESS

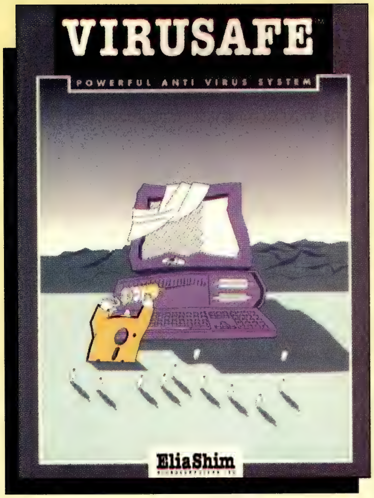
NO SURCHARGE

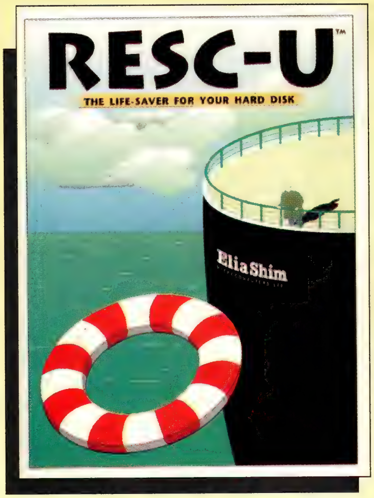
We accept Fortune 500 & Government bids & PO's.
We check for COD and Credit Card fraud!

Prices & specifications subject to change without notice. Shipping charges are nonrefundable. Returns must be in new condition and in original packaging. Defaced items may not be returnable. No refunds on software or special orders (items not listed in ad.) All refunds subject to 15% restocking fee. Some exceptions may apply to unbeatable deals. All trademarks are registered trademarks of their respective companies. Personal checks held for clearance. We reserve the right to refuse any sale for any reason. ©1994 MegaHaus Inc.

Circle 150 on reader service card.

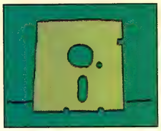
Two DOS Resources That Could Save Your PC!





Anti-Virus

VIRUSAFE: Virus infection can cause irreparable damage to files and programs on your computer. That is why we offer **VIRUSAFE**, the most advanced anti-virus protection system available. **VIRUSAFE** combats virus infection by finding and eliminating thousands of known file and boot sector viruses and their mutations. **VIRUSAFE** can "learn" to recognize and remove unknown viruses and automatically removes Stealth viruses while active in memory.



HD Recovery

RESC-U: Hard disks are known to crash for multiple and diverse reasons. **RESC-U** is a quality recovery system that will help you overcome the sudden, unexpected crash of your hard disk in a matter of minutes. **RESC-U** allows you to create a record of the vital information of your hard disk and save it to a diskette. This record is used to restore damaged information after a crash. With **RESC-U**, you can rest assured that the contents of your hard disk are as secure as in a safe!

EliaShim
Microcomputers, Inc.
4005 Wedgemere Drive, Tampa, FL 33610 TEL: 800-477-5177 FAX: 813-744-5197

Circle 153 on Reader Service Card.

MOTHERBOARDS INTERNATIONAL
by Shambis Corporation
#1 in CUSTOMER SATISFACTION!!!

Your Upgrade Warehouse

PENTIUMS

Intel PCI	$1650
AIR VESA	$1699
TYAN VESA	$1750
Oasis Green VLB	$1727
Super Micro VLB	$1800

Call For Specifications

PCI's

AMI Super Voyager PCI

486DX-33	CALL
486DX2-66	CALL
Vega 4-PCI, 3-VESA, 5-ISA, 512K 486DX266	$877

Call For Other PCI's

DIAMOND

486SX-33 VESA Fastbus	$456
486DX-40 VESA Fastbus	$558
486DX-33 VESA Fastbus	$567
486DX2-66 VESA Fastbus	$765

A.I.R. ☆ Made in U.S.A.

★PENTIUM★ 60MHz ISA, 3-VESA, IDE/IO	**$1699**
486DX-33 8-ISA, 3-VESA, 238-PIN	$487
486DX2-66 8-ISA, 3-VESA, 238-PIN	$712
486DX-33 8-EISA, 3-VESA, 238-PIN	$715
486DX2-66 8-EISA, 3-VESA, 238-PIN	$939
486DX-33 ISA/VESA, SCSI/IDE/IO 256K	$555
486DX2-66 ISA/VESA, SCSI/IDE/IO 256K	$815

AMI ☆ Made in U.S.A.

Super Voyager VLB 486DX-33	$679
Super Voyager VLB 486DX-50	$889
Super Voyager VLB 486DX2-66	$999
Enterprise III EISA VESA 486DX-33	$899
Enterprise III EISA VESA 486DX2-66	$1156
Enterprise IV EISA VESA 486DX2-66	$1165

VEGA ☆ Made in U.S.A.

3/486 w/o CPU	$189
486DX-40 8-ISA, 3-VESA	$498
486DX-33 8-ISA, 3-VESA	$499
486DX2-66 8-ISA, 3-VESA	$659
486DX2-66 EISA, 3-VESA	$885

TYAN

★ PENTIUM ★

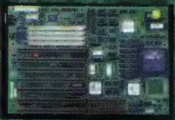
8-16BIT, 3-32BIT, VESA, 2mg Cache, up to 128mg.	**$1950**
MINI 486DX-33 7-16, 2-VLB, 256K	$499
MINI 486DX2-66 7-16, 2-VLB, 256K	$719
EISA/VL 486DX-33 8-32, 2-VLB, 256K	$699
EISA/VL 486DX2-66 8-32, 2-VLB, 512K	$1059

NICE - MCCI ☆ Made in U.S.A.

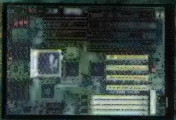
486DX-33 8-ISA, 2-VESA LB 256K	$489
486DX2-66 8-ISA, 2-VESA LB 256K	$679
486DX-33 5-ISA, 3-EISA, 2-VESA 256K, P-24T READY	$559
486DX2-66 5-ISA, 3-EISA, 2-VESA 256K, P-24T READY	$787
486DX-33 8-EISA, 1VESA 512k, up to 256 MEG MEM	$713
486DX2-66 8-EISA, 1VESA 512k, up to 256 MEG MEM	$936

30-DAY GUARANTEE

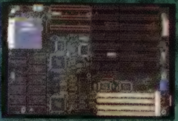
486DX2



486-PCI



PENTIUM



MOTHERBOARD INSTALLATION VIDEO $15

RAM

Please Call for Current Pricing on Memory

1x9 ------------1MB
1x3 ------------1MB
4x9 ------------4MB
1x36-----------4MB
2x36-----------8MB
4x36 ---------16MB

WE ALSO STOCK

Keyboards, CD ROMs, Sound Cards, Network Cards, Floppy Drives, Modems, Tape Drives, and **MORE!**

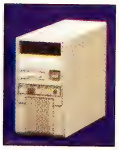
CALL!

1-800 499-3970

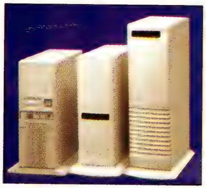
UPGRADE ITEMS

PRO CASES

DC-61- $48 • DC-51 - $65



DC-250 - $69



T2 - $89 • T3 - $85 • T5 - $125

All Cases Come with Mounting Hardware and 230 Watt Power Supply - 250 & 300 Watt Power Supplies are Optional.

Video Cards

Diamond 24 HiColor 1MB VRAM	$119
Diamond Stealth VLB 1MB VRAM	$165
Diamond Viper 2MB VRAM	$365
ATI UltraPro 2MB VRAM	$359
Orchid Fahrenheit S3	$165
Hercules Dynamite Pro	$165
STB Pegasus 2MB VRAM	CALL
STB Pegasus 4MB VRAM	CALL

MONITORS

Viewsonic 14″, 15″, 17″, & 20″	CALL
MAG 15″, & 17″	CALL
CTX 14″, 15″, 17″, & 21″	CALL
NEC 14″, 15″, & 17″	CALL
Proton Black 14″, 15″, 17″, & 21″	CALL

HARD DRIVES

Quantum	CALL
Toshiba	CALL
Fujitsu	CALL
Western Digital	CALL
Maxtor	CALL
Seagate	CALL

CONTROLLERS

Adaptec	CALL
Ultrastor	CALL
Promise	CALL
Buslogic	CALL
DTC	CALL

Yes! You can purchase motherboards w/o CPU!

We Have Benchmark Tests on All Motherboards. Let Us Compare.

MOTHERBOARDS INTERNATIONAL
by Shambis Corporation
8361 E. Evans Road #110 • Scottsdale, AZ 85260
1.800.499.3970 • In Arizona 602.596.5226 • Fax 602.596.1919

Circle 155 on Reader Service Card

DRG MARKETPLACE

READERS: To help you learn more about the many interesting products available for your PC, **DOS Resource Guide** has created the **DRG MARKETPLACE**. This special section makes it easy to contact companies offering a wide range of accessories, software, and services. Listings are grouped by product category for your convenience.

VENDORS: Listings in the **DRG MARKETPLACE** are sold by the column inch. *For more information or to place your ad, contact Linda Guyette or Laurie Donohue at* ***800-343-0728*** *or* ***603-924-0200.***

HARDWARE

COMPUTER SYSTEMS

ASCI COMPUTER

Helping You Put It All Together

BIG DISCOUNTS

FAST DELIVERY

CASES • MOTHERBOARDS
RAM • HARD DRIVES
FLOPPY DRIVES • CD-ROM
I/O CARDS • MODEMS
MICE • KEYBOARDS
MONITORS • MULTIMEDIA
SCANNERS
AND MUCH MORE. . .

COMPLETE SYSTEMS AT DISCOUNT PRICES

$999.00

SYSTEM SPECIAL: Intel® 486SX-25 VLB/ZIF, 420Mb HDD, 4Mb RAM, 3.5" FDD, VLB IDE, VLB 1Mb SVGA, Keyboard, Mouse, 14" Mono VGA Monitor, DOS 6.2, 2S/1P/1G Port, DeskTop or Mini-Tower Case, UPGRADEABLE!!!

800-238-ASCI

FAX: 516-935-5231

Call Today for Your

FREE CATALOG

Circle 134 on Reader Service Card

HARDWARE

VOICE RECOGNITION

THRIFTY TALKER

Real/synthetic speech, music
-- from parallel printer port.
LAP OR DESKTOP, PC/XT ON UP
-- *with or without hard disk!*

$19.95 for:

LPT DAC, AM pocket radio (acts as amplified speaker), and lots of software, manuals. Program is compiled BASIC. Annunciators, language, songs. For fun or profit.
360K bytes of speech in RAM
-- *even on an old PC/XT!*

MC/VISA or check (allow for clearance). Add $3.50 S&H.
Get two for one S&H charge.
Or, get three, and we pay shipping!

THE THRIFTY TALKER
3205 SE Spyglass Drive
Vancouver, WA 98684
FAX: 206-260-1660

Circle 131 on Reader Service Card

Say you saw it in the DRG Marketplace

SOFTWARE

UTILITIES

TURN ANY PRINTER INTO A PERSONAL PLOTTER

PRINTAPLOT® 3.0

"...it is unbeatable,"... CADalyst 2/93

Convert HPGL files
Change pen widths, patterns, colors
Tile "A" to "E" size drawings
Print continuous long plots
Adjust scaling & rotation
Pops up inside any graphics application
20 times faster than a pen plotter
Supports over 1,200 printers: no jaggies
Free technical support

$99

1-800-825-4115

insight Development Corp. 510-244-2000 • 510-244-2020 (FAX)
2420 Camino Ramon, Suite 205, San Ramon, CA 94583

Circle 147 on Reader Service Card

PEN PLOTTER EMULATOR

FPLOT turns your printer into an HP pen plotter. Fast hi-res, no jagged lines. Vary line width, color. Screen preview--zoom, pan. Works with most CAD programs. Supports most printers. Requires DOS 2.1 or higher. $119.00 (plus $3.00 S&H). Visa/MC/Check/MO accepted.

FPLOT Corporation
24-16 Steinway Street, Suite 605
Astoria, NY 11103
(718) 545-3505

Circle 143 on Reader Service Card

EDOS

Enhanced DOS for Windows

"Best Windows Utility '93...Utilities you can't do without," PC World.
"Top 5 Utilities for '93," PC Magazine.

- Add 64-96K to a DOS session
- Unique Menu Bar w/tools
- Extended clipboard support
- Quick/Smart session exiting
- Enhanced PIF editor

Only $69.95. For brochure or to order, call **(800) 248-0809**. Or, try shareware version: download **BBS 503-694-2220 (N81).**

Circle 145 on Reader Service Card

HARDWARE

MEMORY/UPGRADES

AmRam

408-975-2440
FAX 408-975-2446

70ns 1Mx9 $39

TYPE	PKG.	100ns	80ns	60ns
256K x 4	DIP	$5.83	$5.96	$5.99
1M x 1	DIP	$5.00	$6.00	$6.00
256K x 9	SIMM	$9.99	$12.50	$13.99
		80ns	70ns	60ns
1M x 9 3 chip	SIMM	$41.00	$39.00	$43.00
1M x 9 9 chip	SIMM	$44.00	$45.00	$46.99
4M x 9 9 chip	SIMM	$153.00	$153.00	$161.00
1M x 36 72pin	SIMM		$165.00	
4M x 36 72 pin	SIMM		$669.00	

SIPPS AND CACHE MEMORY IN STOCK!

70ns 4Mx9 $153

NOW AVAILABLE! HARD DRIVES

ST-3290A	261MB	14ms	IDE	$224
ST-3391A	341MB	12ms	IDE	$285
CFA-540A	540MB	10ms	IDE	$389
ST-12550N	2.1GB	9ms	SCSI2	$1661

OTHER MODELS IN STOCK!

MATH CO-PROCESSORS

83SX87	16-33	CYRIX	$56
83DX87	33MHZ	CYRIX	$55
83D87-40	40MHZ	CYRIX	$61

LOW WAREHOUSE PRICES!

2059 Camden Avenue, #330 San Jose, CA

All Merchandise 100% guaranteed. We accept VISA, MC, Discover, CODs and company PO's with prior approval. Price, delivery & manufacturer subject to change. Minimum order $100. Returns require RMA; non defective returns subject to a 15% restocking fee.

Did You Know ...

68% of DOS Resource Guide readers use Windows.

SECURITY

Discover who just read your private E-mail! Send for your copy of our **FREE** information security report. Don't delay, write today!

DataSafe Computing
Dept F, PO Box 22508
Alexandria, VA 22304

Circle 130 on Reader Service Card

CONVERT & COMPARE

with

Word for Word

The award-winning **Word For Word** allows you to instantly convert word processing, spreadsheet, database and compound documents between Mac, DOS and Windows formats. Over 15,000 conversion combinations!

- **Quick & Easy Conversions** - Converts more than 3 pages per second.
- **Compound Document Support**
- **Retain Formatting & Layout**
- **Automatic Recognition** - Identifies many Source Formats of your documents.
- **Windows File Viewer** - View and print files without converting them and without the originating application.

List Price: $149.00.

DocuComp

This document comparison and redlining software allows you to easily compare any two versions of a document or files from different word processors. Track inserted, deleted, replaced and moved text. Unique comparison window shows the changes between the versions, lets you scroll through them synchronously. Can create a redlined document using a variety of formats to show the changes. Print a Comparison Summary or Revision List that lists every change by page and line number showing the actual modified text. Save hours of review time! A must for contracts, or any documents that are revised often. Available in DOS, Mac and Windows.

List Price: $179.00.

Limited Time Offer: $99 ea.

MASTERSOFT **800-624-6107**

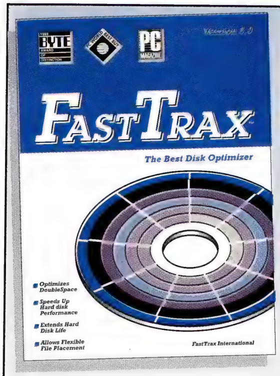
8737 E. Via de Commercio
Scottsdale, AZ 85258

Fax 602-948-8261

* Offer good through August 1, 1994

Circle 121 on Reader Service Card

Say you saw it in
DOS Resource Guide



The Best Disk Optimizer

Version 5.0

- Improves DOS, Windows and OS/2 FAT File System performance. Extends disk life
- Allows custom positioning of files and free space with no limits on size or number of files
- Places files within disk cylinders, reducing track-to-track seeks
- Defragments DoubleSpace drives 5-20 times faster than DOS
- Includes directory sorting, surface testing and repair, and batch operation
- $70 new users, $30 upgrade users, $40 competitive users
- Requires DOS and 2MB of RAM

"For the most finely tuned fragmentation available choose FastTrax...it's the only program that lets you organize your disk exactly the way you want it."

Reprinted from "PC Magazine", September 14, 1993
Copyright © 1993, Ziff-Davis Publishing Company L.P.

PC MAGAZINE

PC WORLD BEST BUY

"Being able to customize optimizations to fit your applications is what makes the difference...FastTrax wins hands down. It's the most fully featured optimizer around."

FastTrax International
880 Ensenada Avenue Berkeley, CA 94707

24 hour Sales & Tech Support **(510) 525-3510**

Circle 127 on Reader Service Card

Adapt DOS to your special needs

Be more productive with MSI enhancements for your standard DOS PC. Our HandiWARE™ software products offer improved access to persons with physical challanges. Our MAGic™ screen magnification software provides instant display from 2x to 12x for low vision needs. And our newest product --TEAM-- provides complete environment control at very low cost. Call us today for more detailed information.

Microsystems Software, Inc.

600 Worcester Road • Framingham, MA 01701
Toll Free: (800) 828-2600 • FAX: (508) 879-1069
BBS: (508) 875-8009

MSI™

Circle 119 on Reader Service Card

HAVE WE GOT A DEAL FOR YOU!!

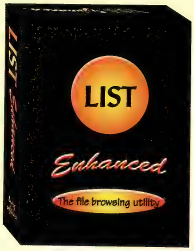
A Great BASIC COMPILER for only $39.95 !!

- **Create your own stand-alone programs that run incredibly fast!**

See page 73 for details.

VIEWING PLEASURE

FOR YOUR



Search up to 80MB/minute
Requires less than:
40K disk & 128K RAM
Up to 6 video modes in LIST
including 132x25 & 132x44
View, print, add & Extract
files from archived files
Split-screen viewing windows
"Sweep" feature executes a
program on all tagged files
Network compatible
WINDOWS COMPATIBLE

$99

Multi-User Discounts and Site Licensing Available Call for Further Information

BUERG SOFTWARE

Phone: (707) 769-LIST FAX: (707) 769-5479
850 Petaluma Blvd. North, Petaluma CA 94952

Circle 109 on Reader Service Card

SOFTWARE

POWERBASIC TOOLS

Boost Your BASIC with New ProBas™ 5.6!

New ProBas 5.6 boosts your Basic programs into orbit! And no royalties, ever!
- Over 900 routines, most in assembly for speed & size! Loads of examples!
- Arrays, Comms, Strings, Equip, Files, DOS, Memory, Sorts, F/Xs, Virtual windows, GUI,...
- 1200 pages of docs with detailed tuorial!
- Fast, flexible code for professional programs!
- Unlimited free tech support! **Just $195!**

Risk FREE 60 day trial!

More QuickBasic, Power Basic, Visual Basic & C tools:

ProMath™ 2.0 - 240 math/engineer/stat/physics/graph fns	$149
FinLib™ 1.1- interest/bond/cashflow/annuity calculations	$149
ProFont™ 2.1- screen fonts - rotate/color/size/edit/input	$125
ToolKit™ 3.1- menus, dialogs, B-Tree, EXE patch	$125
ProVoice™ 1.0- telephone voice processing library	$199
TeleComm™ 1.2 - modem transfer & terminal emulation	$75
SpellCheck™ 1.1 - spelling checking library & dictionary	$49
Ace Comparator™ 1.2- intelligent program diff display	$99

FREE demo disk & booklet - call:
800-447-9120 ext. 1089

TeraTech - Tools for Programmers™
Dept 1087, 100 Park Ave, Suite 360, Rockville MD 20850 USA
Int'l: +1-301-424-3903 Fax: 301-762-8185 BBS: 301-762-8184 9600-N-8-1

Copyright TeraTech 1994. All rights reserved. Trademarks are the property of their respective holders.

Circle 114 on Reader Service Card

SOFTWARE

PUBLIC DOMAIN/SHAREWARE

DATA MATE

follow our star...

FREE SHAREWARE CATALOG

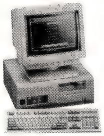
Over 3400 high quality, low priced programs + bonuses + free disk offers.
CD-ROMS too, ask for our menu/price list.

DATA MATE
24 Crestwood Lane, Dept. D
Lake Ronkonkoma, NY 11779

Circle 126 on Reader Service Card

MISCELLANEOUS

BUSINESS OPPORTUNITIES



FREE CBSI 486 *Business Computer*

Earn $4,000 Per Month From Your Home With A Computer!

Quit spending money on your computer and let it earn money for you. This is a proven turnkey business an individual or couple can run. If you purchase our software and business program, we will give you the computer and printer. If you already own a computer, you may receive a discount. Begin part-time and still retain the security of your present position. Financing available. We accept AMEX, MC, Visa and Discover.

To receive your free cassettes and color literature, call toll-free:

1-800-343-8014, ext. 73
Computer Business Services, Inc.

Circle 146 on Reader Service Card

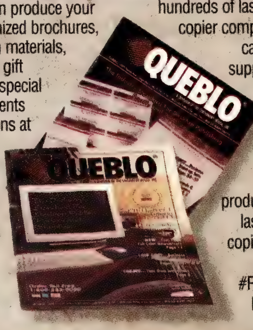
MISCELLANEOUS

SUPPLIES

FREE CATALOG

Now you can produce your own customized brochures, presentation materials, newsletters, gift certificates, special announcements and invitations at a fraction of the expense a commercial printer would charge. Our catalog features hundreds of laser printer and copier compatible papers, card stocks and supplies designed to create spectacular promotional materials.

All Queblo products are 100% laser printer and copier compatible. Request code #R1028 for your FREE Catatlog.



Call Toll Free 1-800-523-9080

QUEBLO® 1000 Florida Avenue, Hagerstown, Maryland 21741
24 Hour Fax 1-800-55-HURRY (48779)

Circle 136 on Reader Service Card

ORGANIZE AND PROTECT YOUR COPIES OF DOS RESOURCE GUIDE

Now there's an easy way to keep copies of your favorite magazine readily available for future reference. Designed exclusively for DOS Resource Guide by Jesse Jones Industries, these custom-made titled cases and binders are sized to hold a year's issues (may vary with issue sizes)
- Reinforced board covered with durable leather-like material in red • Title is hot-stamped in gold
- Free personalization foil included for indexing year • Cases are V-notched for easy access
- Binders have special spring mechanism to hold individual rods which easily snap in.

DOS Resource Guide
Jesse Jones Industries, Dept. DOS
499 East Erie Avenue, Philadelphia, PA 19134

Quantity	Cases	Binders
One	$ 7.95	$ 9.95
Three	$ 21.95	$27.95
Six	$ 39.95	$52.95

Add $1 per case/binder postage and handling. Outside USA $2.50 per case/binder. (U.S.funds only.)

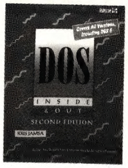
Please send______cases;______binders for DOS Resource Guide Magazine.
☐ Enclosed is $__________.
☐ Charge my: (Minimum $15)
☐ American Express ☐ Visa ☐ MasterCard ☐ Diners Club
Card #______________________________Exp. Date ______________
Signature ______________________________

Charge Orders: Call TOLL FREE 7 days, 24 hours ☎ 1-800-825-6690

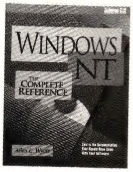
Name ______________________________
Address ______________________________
City/State/Zip ______________________________

SATISFACTION GUARANTEED

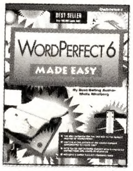
4335H-XX $39.95
Counts as 2/Hardcover



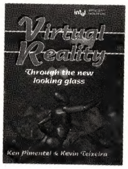
881918P-XX $27.95
Counts as 2



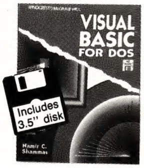
881832XP-XX $29.95
Counts as 2



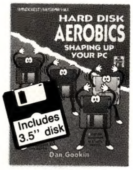
8818958P-XX $24.95
Counts as 2



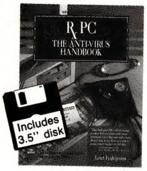
4196P $22.95



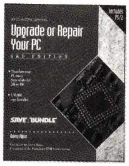
4366P $29.95



4380H-XX $39.95
Counts as 2/Hardcover



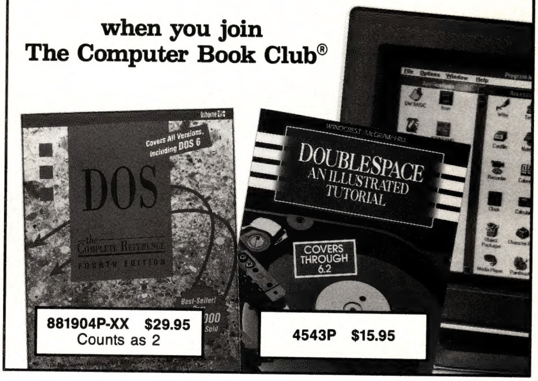
4262H $39.95
Hardcover



4269P $19.95

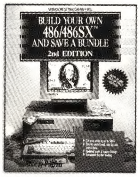
Select any 5 books for only 4^{95} (Values to $176.75)

when you join The Computer Book Club®

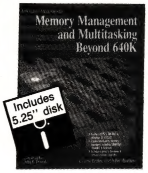

DOS the Complete Reference FOURTH EDITION
Covers All Versions, Including DOS 6
DOUBLESPACE AN ILLUSTRATED TUTORIAL
COVERS THROUGH 6.2

881904P-XX $29.95
Counts as 2

4543P $15.95

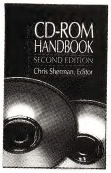


4270H $29.95
Hardcover

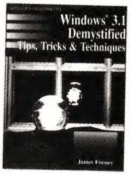


4069P-XX $29.95
Counts as 2

3279P $26.95

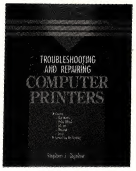


056693H-XX $70.50
Counts as 2/Hardcover

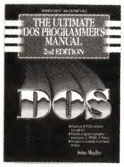


4180P $24.95

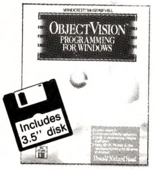
4379H-XX $44.95
Counts as 2/Hardcover



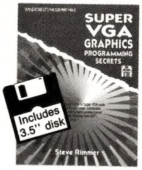
3923P $24.95



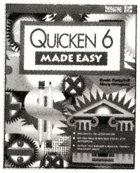
4221H-XX $40.00
Counts as 2/Hardcover



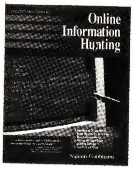
4258H-XX $34.95
Counts as 2/Hardcover



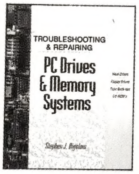
4413H-XX $49.95
Counts as 2/Hardcover



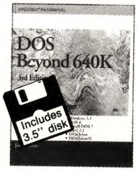
881890P $19.95



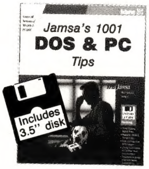
3943P $19.95



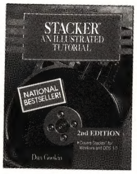
4491H $39.95
Hardcover



4356H $39.95
Hardcover



881821P-XX $39.95
Counts as 2



4447P $19.95

As a member of The Computer Book Club...

...you'll enjoy receiving Club bulletins every 3-4 weeks containing exciting offers on the latest books in the field at savings of up to 50% off of regular publishers' prices. If you want the Main Selection do nothing and it will be shipped automatically. If you want another book, or no book at all, simply return the reply form to us by the date specified. You'll have at least 10 days to decide. If you ever receive a book you don't want, due to late mail delivery of the Bulletin, you can return it at our expense. And you'll be eligible for *FREE* BOOKS through the Bonus Book Program. Your only obligation is to purchase 3 more books during the next 2 years, after which you may cancel your membership at any time.

All books are softcover unless otherwise noted. Publishers' prices shown. If you select a book that counts as 2 choices, write the book number in one box and XX in the next. A shipping/handling charge & sales tax will be added to all orders. ©1994 CBC

The Computer Book Club®
Blue Ridge Summit, PA 17294-0820

YES! Please rush me the books indicated below for just $4.95 plus shipping/handling and applicable sales tax. Enroll me as a member of **The Computer Book Club** according to the terms outlined in this ad. If not satisfied I may return the books within 10 days without obligation and my membership will be cancelled.

If you select a book that counts as 2 choices, write the book number in one box and XX in the next.

Name ____________________

Address ____________________

City/State ____________________

Zip __________ Phone ____________________

Offer valid for new members only, subject to acceptance by CBC. Canada *must* remit in U.S. funds drawn on U.S. banks. Applicants outside the U.S. and Canada will receive special ordering instructions. A shipping/handling charge & sales tax will be added to all orders.

DOS894

SHAREWARE EXCHANGE

A Handy Batch Of Batch Enhancers

by Hardin Brothers

Last issue, almost all the submissions to this column were utility programs. For this issue, almost everyone nominated a batch-file menu system. Are you folks collaborating before you send in your recommendations?

Each of the three programs discussed here takes a different approach to the problem of creating attractive menus in batch files. I don't know whether any of the three qualifies as the best, because each performs in a unique way. If you decide to use one of these menu programs, make your choice based on your own programming requirements, not on some type of "good/better/best" rating system.

Three programs take different approaches to the problem of creating attractive menus in a batch file.

Actually, you don't really need a special program to create attractive batch-file menus. If you use DOS 6.x and its new CHOICE command, you already have most of the tools you need for the job. (See the feature "The Secrets of Top-Notch Batch-File Menus" in this issue, page 39.) Your menus, however, will take a lot of work to create; they won't have moving cursor bars or mouse support; and they'll appear on screen slowly. By contrast, these shareware programs all display menus immediately; let you test color combinations, background characters, and other special effects with little effort; and generally give your batch-file programs a professional look.

Slick Trickster

The first nomination comes from Jerry Solomon of Boca Raton, Florida:

> ***BATMNU 2.3** is a slick program that includes the ability to run programs directly from a menu, which speeds up operation, since batch files don't need to do errorlevel checking after the user makes a choice. It's well documented by the author.*

BATMNU is indeed slick. When you're ready to display a menu, you include a single line in your batch file that launches BATMNU and tells the program which file to search for the menu, where to start in that file, and which choice in the menu is the default.

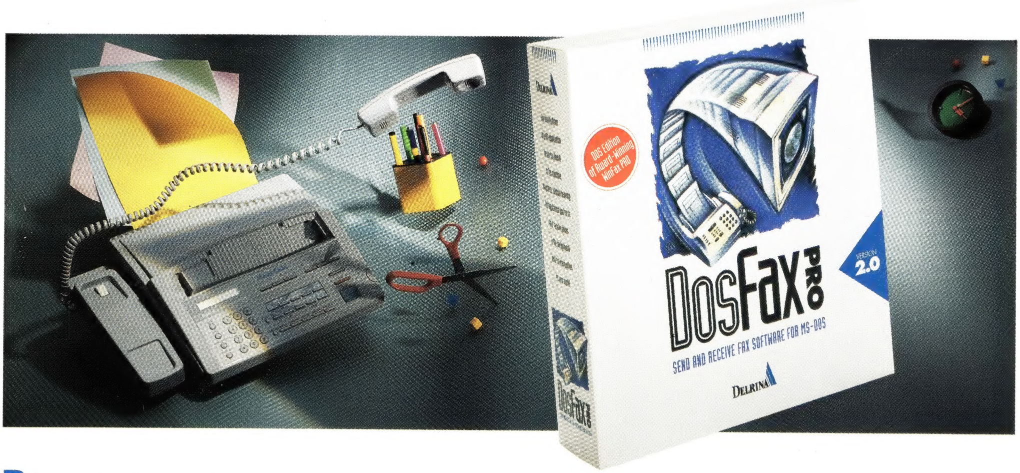
Within the menu itself, you can define a "header" for the menu (information that appears at the top of the screen), colors and fill characters for the screen window, each menu option and the help text that should appear when the cursor is on that option, and a footer for the menu. You can enable or disable mouse support and include hot keys if you choose.

The user can move around the menu, which he or she can display in either one or two columns, with a cursor bar, by pressing hot keys, or with a mouse. Once the user makes a selection, BATMNU can either run the program directly (without returning to your batch program) or place the selection in an environment variable of your own choosing.

BATMNU includes a couple of other interesting features. You can place the menu definition either in your batch file or in a separate file. You can include multiple menus in a single file. If you use a separate file, you can give it a hidden attribute so that it's somewhat protected from other workers in your office. A special version that reads encrypted menu files is available to registered users.

(Continued on page 76)

Get Delrina DosFax PRO™ 2.0 and put faxing where it really belongs—on your PC.


DOS Edition of Award-Winning WinFax PRO
Version 2.0
DosFax PRO
Send and receive fax software for MS-DOS
Delrina

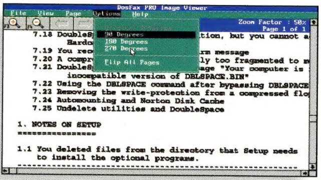
Discover just how much faster and easier it is to send and receive your faxes directly from your PC with Delrina DosFax PRO™. Based on Delrina WinFax PRO™, the world's best-selling and most award-winning fax software, DosFax PRO makes it easy to send, receive and manage your faxes right from the comfort and privacy of your own PC.

Now, sending a fax directly from within any DOS application is as easy as printing a document. All you need is DosFax PRO and your fax modem.

Now Faxing Is as Easy as Printing

DosFax PRO makes sending out complex faxes a cinch. You can easily attach multiple documents onto a single outgoing fax – a memo from your word processor, numbers from your spreadsheet and illustrations from your drawing package. And you control how you receive faxes, either in the background so your work isn't interrupted or directly to your printer. DosFax PRO even makes recording and managing names and fax numbers simple with built-in phonebooks.

> "If you have an external or internal modem that can send or receive faxes, DosFax PRO 2.0 is the package for you."
>
> *Dos Resource Guide/ May 1994*



```
    7.18 DoubleS                              tion, but you cannot a
         Hardc
    7.19 You rec                              rm message
    7.20 A compr                              ly too fragmented to m
    7.21 DoubleS                              age "Your computer is
         incompatible version of DBLSPACE.BIN"
    7.22 Using the DBLSPACE command after bypassing DBLSPACE
    7.23 Removing the write-protection from a compressed flo
    7.24 Automounting and Norton Disk Cache
    7.25 Undelete utilities and DoubleSpace

 1. NOTES ON SETUP
 ================

 1.1 You deleted files from the directory that Setup needs
     to install the optional programs.
 ------------------------------------------------------------
```

Sending, receiving and managing faxes can be easier and more convenient with Delrina DosFax PRO.

All Your Fax Information at Your Fingertips

And you won't get caught scrambling through piles of paper when you're in a rush to find that important fax because DosFax PRO keeps a detailed record of all the faxes you send and receive. And unlike any other fax software for DOS, DosFax PRO allows you to view your faxes on screen, rotate documents which have been faxed upside down, print fax documents to your printer, and export fax images.

So get DosFax PRO for only $49 (US) and put faxing where it really belongs – on your PC. But remember this is a limited-time offer, so take advantage of this **new low-price** while you can by calling us directly.

DELRINA®
1-800-268-6082

DosFax PRO, WinFax PRO and Delrina are trademarks of Delrina (Canada) Corporation. All other product names are trademarks of their respective owners. ©1994 Delrina (Canada) Corporation. All rights reserved.

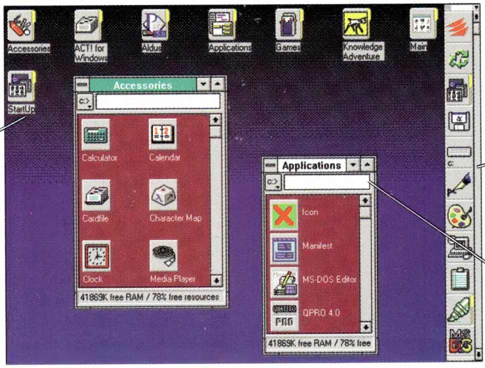
Circle 160 on Reader Service Card

SideBar™ Wonks Your Windows to Perfection

SideBar eliminates the last bit of DOS clunkiness from MS Windows.

Object-oriented interface makes it easier to manage files, disk drives and printers.

Simply drag and drop files to copy, move or print them.



Now opening, copying, moving, deleting, renaming and printing are intuitive.

Speed up the way you work by customizing SideBar to conserve desktop space.

Sometimes prefer to key-in DOS commands? The command line's always there.

SideBar is the efficient desktop for Windows—a small, fast shell that automatically creates objects that make it simpler to manage applications, documents, folders, drives, TSRs, printers, etc. All you need to do is point and click.

SideBar simply makes Windows more 'windows-like.' And that makes your work go quicker and easier. It's an idea incorporated in Microsoft Office, but we've taken it to its obvious conclusion.

SideBar makes MS Windows a better place to work.

Quarterdeck

Quarterdeck Office Systems, 150 Pico Boulevard, Santa Monica, CA 90405 (310) 392-9851 Fax (310) 314-4219
Quarterdeck International Ltd., B.I.M. House, Crofton Terrace, Dun Laoghaire Co. Dublin, Ireland Tel.(353) (1) 284-1444 Fax: (353) (1) 284-4380

©1993 Quarterdeck Office Systems. Trademarks are property of their respective owners.

Circle 107 on Reader Service Card